

**ANALISIS KINERJA OPENFLOW *BASED SWITCH* PADA  
ROUTER LINKSYS WRT54GL DAN TP-LINK TL-WR1043ND  
DALAM ARSITEKTUR *SOFTWARE DEFINED NETWORKING***

**SKRIPSI**

**KEMINATAN TEKNIK KOMPUTER**

untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Yasin Rizqi Afandi  
NIM: 135150301111129



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018**

**PENGESAHAN**

ANALISIS KINERJA OPENFLOW BASED SWITCH PADA ROUTER LINKSYS WRT54GL  
DAN TP-LINK TL-WR1043ND DALAM ARSITEKTUR SOFTWARE DEFINED  
NETWORKING

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer


Disusun Oleh :  
Yasin Rizqi Afandi  
NIM: 13515030111129

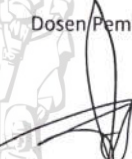
Skripsi ini telah diuji dan dinyatakan lulus pada  
Tanggal 01 Agustus 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

  
Sabriansyah Rizqika A., S.T., M.Eng.  
NIP. 19820809 201212 1 004

  
Widhi Yahya, S.Kom, M.Sc.  
NIK. 2016078911211001

Mengetahui  
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP. 19710518 200312 1 001

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata di dalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 08 Agustus 2018



Yasin Rizqi Afandi

NIM: 135150301111129

## KATA PENGANTAR

Puji syukur peneliti panjatkan kepada Tuhan Yang Maha Esa atas limpahan rahmat dan petunjuk-Nya, sehingga peneliti dapat menyelesaikan skripsi yang berjudul “ANALISIS KINERJA OPENFLOW BASED SWITCH PADA ROUTER LINKSYS WRT54GL DAN TP-LINK TL-WR1043ND DALAM ARSITEKTUR SOFTWARE DEFINED NETWORKING” dengan baik.

Dalam penyusunan dan penelitian skripsi ini tidak lepas dari bantuan moral dan materiil yang diberikan dari berbagai pihak, maka peneliti mengucapkan banyak terima kasih kepada:

1. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D. selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
2. Bapak Heru Nurwarsito, Ir., M.Kom. selaku Wakil Ketua I Bidang Akademik Fakultas Ilmu Komputer Universitas Brawijaya Malang.
3. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D. selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya Malang.
4. Bapak Sabriansyah Rizqika Akbar, S.T, M.Eng. selaku Ketua Program Studi Teknik Komputer Universitas Brawijaya Malang dan Dosen Pembimbing I yang telah memberikan pengarahan dan bimbingan kepada peneliti, sehingga dapat menyelesaikan skripsi ini dengan baik.
5. Bapak Widhi Yahya, S.Kom, M.Sc. selaku Dosen pembimbing II yang telah memberikan pengarahan dan bimbingan kepada peneliti, sehingga dapat menyelesaikan skripsi ini dengan baik.
6. Kedua orang tua dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian, dan kesabarannya memberikan semangat kepada peneliti, serta senantiasa tiada hentinya memberikan doa demi terselesaikannya skripsi ini.



7. Seluruh civitas akademika Informatika Universitas Brawijaya dan terkhusus untuk teman-teman Teknik Komputer Angkatan 2013 yang telah banyak memberi bantuan dan dukungan selama peneliti menempuh studi di Teknik Komputer Universitas Brawijaya dan selama penyelesaian skripsi ini.
8. Seluruh pihak yang tidak dapat diucapkan satu persatu, peneliti mengucapkan banyak terima kasih atas segala bentuk dukungan dan doa sehingga laporan skripsi ini dapat terselesaikan.

Peneliti menyadari bahwa dalam penyusunan skripsi ini masih terdapat banyak kekurangan, sehingga saran dan kritik yang membangun sangat peneliti harapkan. Akhir kata peneliti berharap skripsi ini dapat membawa manfaat bagi semua pihak yang membutuhkannya.

Malang, 08 Agustus 2018

Yasin Rizqi Afandi.

Ysins.mail@gmail.com



## ABSTRAK

Pada arsitektur software defined network (SDN) terdapat sebuah protokol yang bersifat open source yang disebut dengan Openflow. Openflow dapat ditambahkan pada open interface device. Router Linksys WRT54GL dan TP-Link TL-WR1043ND dapat menggunakan Openflow dengan menginstal firmware open interface seperti Openwrt. Beberapa penelitian sebelumnya telah dilakukan pengujian terkait kinerja router Linksys WRT54GL dan TP-Link TL-WR1043ND berdasarkan parameter sumberdaya jaringan seperti throughput, round trip time dan jitter. Pada penelitian ini dilakukan pengujian kinerja protokol Openflow pada arsitektur SDN dengan menambahkan sumberdaya komputasi sebagai parameter uji. Parameter sumberdaya komputasi meliputi penggunaan sumberdaya CPU dan memory sedangkan, parameter jaringan meliputi round trip time, throughput dan waktu pembentukan flow table. Berdasarkan seluruh hasil pengujian, penambahan Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND yang diinstal Openwrt, dapat meningkatkan penggunaan sumberdaya CPU dan memory serta menurunkan kinerja jaringan seperti round trip time dan throughput. Penurunan kinerja router tersebut disebabkan karena penambahan proses Openflow pada router yang menyebabkan turun nya kinerja switching dan mekanisme dari hybrid Openflow switch yang menyebabkan penambahan waktu pemrosesan forwarding data.

Kata kunci: *software define network, Openflow, tp-link tl-wr1043nd, linksys wrt54gl*

## ABSTRACT

*In software defined network architecture (SDN) there is an open source protocol called Openflow. Openflow can be added to the open interface device. The Linksys WRT54GL and TP-Link TL-WR1043ND routers can use Openflow by installing open-source firmware such as Openwrt. Several previous studies have tested the performance of Linksys WRT54GL and TP-Link TL-WR1043ND routers based on network resource parameters such as throughput, round trip time and jitter. This study is conducted testing the performance of the Openflow protocol on the SDN architecture with adding parameters of computational. Computational parameters include the use of CPU and memory resources then, network parameters include round trip time, throughput and flow table formation time. Based on all test results, the addition of Openflow on the Linksys WRT54GL and TP-Link TL-WR1043ND router who installed openwrt, can increase the use of CPU and memory resources and can degrade network performance such as round trip time and throughput. The decrease performance of router is due to the addition of the Openflow process on the router who cause decreased the switching performance and the mechanism of the Openflow hybrid switch who cause increased forwarding data processing time*

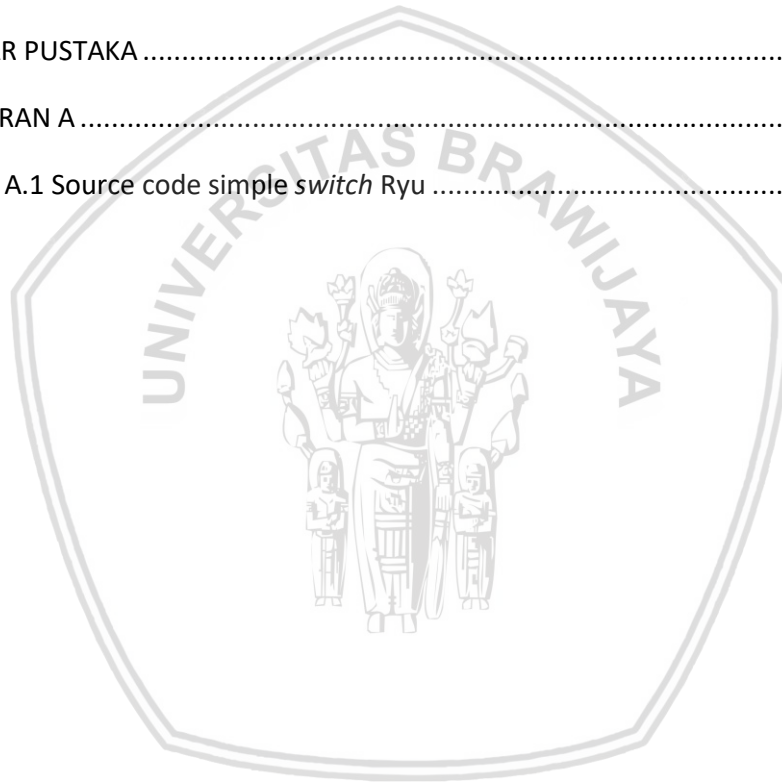
**Keywords:** *software define network, Openflow, tp-link tl-wr1043nd, linksys wrt54gl*

## DAFTAR ISI

PENGESAHAN .....	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS .....	Error! Bookmark not defined.
KATA PENGANTAR .....	iii
ABSTRAK .....	v
ABSTRACT .....	vi
DAFTAR ISI .....	vii
DAFTAR GAMBAR .....	x
DAFTAR TABEL .....	xii
DAFTAR LAMPIRAN .....	xiv
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Manfaat .....	3
1.5 Batasan Masalah .....	3
1.6 Sistematika Pembahasan .....	3
BAB 2 LANDASAN KEPUSTAKAAN .....	5
2.1 Kajian Pustaka .....	5
2.2 Dasar Teori .....	6
2.2.1 <i>Software Defined Network</i> .....	6
2.2.2 Openflow .....	8
2.2.3 Ryu .....	9
2.2.4 Router .....	9
2.2.5 Openwrt .....	11

2.2.6 Arsitektur <i>Switch</i> .....	11
2.2.7 <i>Latency</i> .....	15
2.2.8 Throughput.....	16
BAB 3 METODOLOGI .....	17
3.1 Studi Literatur .....	18
3.2 Analisis Kebutuhan.....	18
3.2.1 Kebutuhan Fungsional .....	18
3.2.2 Kebutuhan Non Fungsional .....	18
3.3 Perancangan dan Implementasi Lingkungan Pengujian .....	21
3.3.1 Perancangan.....	21
3.3.2 Implementasi lingkungan pengujian .....	21
3.3.3 Pengujian.....	22
3.4 Hasil dan Pembahasan .....	22
3.5 Kesimpulan dan Saran.....	22
BAB 4 PERANCANGAN DAN IMPLEMENTASI LINGKUNGAN PENGUJIAN .....	23
4.1 Perancangan .....	23
4.1.1 Perancangan Lingkungan pengujian.....	23
4.1.2 Topologi .....	24
4.2 Implementasi Lingkungan Pengujian .....	25
4.2.1 <i>Control plane</i> .....	25
4.2.2 <i>Data plane</i> .....	26
4.2.3 <i>Aplication plane</i> .....	33
4.3 Pengujian .....	33
4.3.1 Skenario Pengujian dan Pengambilan data .....	34
BAB 5 HASIL DAN PEMBAHASAN.....	38

5.1 Penggunaan <i>memory</i> dan CPU .....	38
5.2 <i>Round trip time</i> .....	42
5.3 <i>Throughput</i> .....	45
5.4 Waktu pembentukan <i>flow table</i> .....	47
BAB 6 KESIMPULAN DAN SARAN .....	49
6.1 Kesimpulan .....	49
6.2 Saran .....	50
DAFTAR PUSTAKA .....	51
LAMPIRAN A .....	53
A.1 Source code simple switch Ryu .....	53





## DAFTAR GAMBAR

Gambar 2.1 Komponen SDN .....	7
Gambar 2.2 <i>Flow</i> entries pada Openflow .....	8
Gambar 2.3 Router Linksys WRT54GL .....	9
Gambar 2.4 Router TP-Link TL-WR1043ND .....	10
Gambar 2.5 Struktur paket <i>switch/router</i> .....	12
Gambar 2.6 <i>Plane</i> pada <i>switch</i> tradisional.....	13
Gambar 2.7 Openflow <i>switch</i> .....	14
Gambar 2.8 Jalur paket Openflow <i>switch</i> .....	14
Gambar 3.1 <i>Flowchart</i> pengerjaan penelitian .....	17
Gambar 3.2 Blok diagram Linksys WRT54GL .....	19
Gambar 3.3 Blok diagram TP-Link TL-WR1043ND.....	20
Gambar 4.1 Skema alur perancangan lingkungan pengujian .....	23
Gambar 4.2 Penggunaan layer pada SDN .....	24
Gambar 4.3 Topologi jaringan pada arsitektur SDN.....	25
Gambar 4.4 Topologi jaringan pada arsitektur tradisional.....	25
Gambar 4.5 Telnet Openwrt .....	27
Gambar 4.6 Contoh data pengujian penggunaan <i>memory</i> dan CPU .....	35
Gambar 4.7 Contoh data pengujian <i>round trip time</i> .....	36
Gambar 4.8 Contoh data pengujian <i>throughput</i> .....	36
Gambar 4.9 Pengujian waktu pembentukan <i>flow table</i> pada <i>controller</i> .....	37
Gambar 5.1 Rata-rata penggunaan <i>memory</i> dan CPU router Linksys.....	40
Gambar 5.2 Rata-rata penggunaan <i>memory</i> dan CPU router Linksys.....	42
Gambar 5.3 Rata-rata <i>round trip time</i> pada router Linksys WRT54G .....	43
Gambar 5.4 Rata-rata <i>round trip time</i> pada router TP-Link WR103ND .....	44

Gambar 5.5 Grafik <i>throughput</i> antara router dan <i>controller</i> pada Linksys WRT54GL dan TP-Link TL-WR1043ND .....	45
Gambar 5.6 Grafik <i>throughput</i> antar <i>host</i> pada Linksys WRT54GL dan TP-Link TL-WR1043ND .....	46



## DAFTAR TABEL

Tabel 2.1 Kajian Pustaka .....	5
Tabel 4.1 Konfigurasi <i>network</i> Linksys WRT54GL.....	28
Tabel 4.2 Konfigurasi <i>network</i> TP-Link TL-WR1043ND .....	29
Tabel 4.3 Konfigurasi Openflow .....	32
Tabel 4.4 Penambahan fungsi waktu pada <i>simple_switch</i> .....	33
Tabel 4.5 Pengujian penggunaan <i>memory</i> dan CPU pada router berdasarkan ukuran <i>file</i> .....	34
Tabel 4.6 Pengujian penggunaan <i>memory</i> dan CPU pada router berdasarkan jumlah <i>host</i> .....	34
Tabel 5.1 Hasil pengujian penggunaan CPU dan <i>memory</i> saat router kondisi idle .....	38
Tabel 5.2 Hasil pengujian penggunaan CPU dan <i>memory</i> pada Linksys WRT54GL berdasarkan ukuran <i>file</i> .....	39
Tabel 5.3 Hasil pengujian penggunaan CPU dan <i>memory</i> pada Linksys WRT54GL berdasarkan jumlah <i>host</i> .....	39
Tabel 5.4 Rata-rata penggunaan <i>memory</i> dan CPU router Linksys WRT54G .....	40
Tabel 5.5 Hasil pengujian penggunaan CPU dan <i>memory</i> pada TP-Link WR1043ND Openflow berdasarkan ukuran <i>file</i> .....	41
Tabel 5.6 Hasil pengujian penggunaan CPU dan <i>memory</i> TP-Link WR1043ND berdasarkan jumlah <i>host</i> .....	41
Tabel 5.7 Rata-rata penggunaan <i>memory</i> dan CPU router TP-Link TL-WR1043ND .....	42
Tabel 5.8 Hasil pengujian <i>round trip time</i> Linksys WRT54GL.....	43
Tabel 5.9 Hasil pengujian <i>round trip time</i> TP-Link WR1043ND.....	44
Tabel 5.10 Hasil pengujian <i>throughput</i> antara router dan <i>controller</i> pada Linksys WRT54GL dan TP-Link TL-WR1043ND .....	45
Tabel 5.11 Hasil pengujian <i>throughput</i> antar <i>host</i> pada Linksys WRT54GL dan TP-Link TL-WR1043ND .....	46

Tabel 5.12 Hasil pengujian waktu pembentukan <i>flow table</i> router Linksys WRT54GL.....	47
Tabel 5.13 Hasil pengujian waktu pembentukan <i>flow table</i> router TP-Link TL-WR1043ND .....	48



## DAFTAR LAMPIRAN

A.1 Source code simple switch Ryu .....	53
---	----



## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

*Software defined networking* (SDN) merupakan sebuah konsep arsitektur jaringan yang bersifat terpusat, dinamis, mudah dikelola dan dimodifikasi serta dapat menyesuaikan dengan kondisi jaringan yang ada. Konsep SDN memisahkan antara fungsi *data plane* dan *control plane* pada jaringan (Bakhshi, 2017). Pada arsitektur tradisional, fungsi *data plane* dan *control plane* menjadi suatu kesatuan dalam perangkat jaringan. *Data plane* atau dikenal juga dengan istilah *forwarding plane*, merupakan suatu fungsi untuk membawa dan meneruskan paket pada jaringan. *Data plane* hanya bekerja sesuai instruksi yang diberikan oleh *control plane*. *Control plane* berfungsi memberikan keputusan pada setiap paket dalam jaringan, apakah paket tersebut dikirim ataupun dibuang. Selain itu *control plane* juga mengelola tabel *routing*, sistem konfigurasi dan *routing* protokol (Kanika, 2016)

Pada arsitektur SDN terdapat beberapa protokol yang mendukung konsep SDN, salah satunya adalah Openflow. Openflow merupakan protokol komunikasi antara *data plane* dan *control plane* pada arsitektur SDN. Beberapa vendor perangkat jaringan seperti Big Switch, NEC, HP dan IBM mendukung penggunaan Openflow. Pada vendor-vendor tersebut, Openflow ditambahkan dalam perangkat jaringan untuk mendukung penggunaan konsep SDN. Beberapa Vendor yang tidak mendukung Openflow menggunakan protokol SDN lain, misalnya Cisco dan Juniper (Doyle, 2013). Sedangkan untuk vendor yang tidak mendukung SDN, dapat menambahkan Openflow secara manual pada *open interface device*. *Open interface device* adalah perangkat jaringan yang mendukung adanya perubahan dan modifikasi sistem operasi didalamnya.

Openwrt merupakan salah satu *firmware* perangkat jaringan yang mendukung penggunaan protokol Openflow. Openwrt adalah sistem operasi berbasis Linux yang digunakan pada *embeded device* khususnya router (Openwrt, 2013). Dengan menginstal Openwrt, router dapat digunakan sebagai *switch* pada arsitektur SDN. Namun, tidak semua router dapat diinstal Openwrt. Pada setiap router terdapat sebuah sistem operasi yang tertanam didalamnya, dimana sistem operasi tersebut berbeda-beda pada setiap vendor router (Mitchell, 2017). Beberapa vendor yang mendukung penggunaan Openwrt pada router adalah Linksys, TP-Link, Mikrotik dll. Namun, penambahan Openflow secara manual pada router masih jarang digunakan dan jarang diterapkan pada arsitektur SDN.

Pada penelitian sebelumnya, penggunaan Openflow pernah diterapkan pada router Mikrotik. Penelitian tersebut membandingkan penggunaan Openflow pada SDN dengan arsitektur tradisional yang tidak menggunakan Openflow. Penggunaan Openflow pada mikrotik dapat dikatakan baik dikarenakan rata-rata gap antara SDN dan tradisional yang tidak terlalu tinggi, yaitu 409 kbps



*throughput*, 0,5 ms *latency* dan -0,4 ms *jitter*(Abdillah, 2016). Selain itu, penggunaan Openflow juga pernah diteliti pada router TP-Link TL-WR1043ND. Router TP-Link TL-WR1043ND sebagai Openflow software *based* dapat digunakan untuk menggantikan dedicated SDN *switch* untuk mengimplementasikan SDN. Performa TP-Link TL-WR1043ND software-*based* mempunyai hasil pengujian *latency* 5,5 ms, *throughput* 9292 kbps dan *jitter* 0,024 (Rikie Kartadie, 2015).

Router Linksys WRT54GL dan TP-Link TL-WR1043ND dapat diinstal sistem operasi Openwrt dan ditambahkan Openflow. Beberapa penelitian yang terkait sebelumnya kebanyakan hanya menguji dari sisi kinerja jaringannya dan terlihat terjadi penurunan kinerja pada beberapa parameter uji. Oleh karena itu pada penelitian ini dilakukan pengujian menggunakan 2 buah router dari 2 vendor yang berbeda yaitu Linksys dengan seri WRT54GL dan TP-Link dengan seri TL-WR1043ND, berdasarkan parameter sumberdaya komputasi dan jaringan. Penelitian ini bertujuan untuk mengetahui kinerja kedua router tersebut sebagai Openflow *based switch* dan mengetahui pengaruh dari penggunaan Openflow pada kedua router tersebut. Parameter sumberdaya komputasi meliputi penggunaan CPU dan *memory*. Sedangkan parameter sumberdaya jaringan meliputi *round trip time*, *throughput* dan waktu pembentukan *flow table*. Melalui penelitian ini dapat diketahui kinerja dan pengaruh penggunaan Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND sebagai SDN *switch*. Sehingga dapat dijadikan sebagai salah satu pertimbangan, dalam implementasi router sebagai SDN switch dengan spesifikasi perangkat yang sejenis.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka dapat penulis simpulkan beberapa rumusan masalah yang berhubungan dengan latar belakang tersebut yaitu sebagai berikut.

1. Bagaimana tahapan dalam merancang lingkungan pengujian kinerja Openflow-*based switch* pada router Linksys WRT54GL dan TP-Link TL-WR1043ND dalam arsitektur SDN?
2. Bagaimana kinerja Openflow-*based switch* pada router Linksys WRT54GL dan TP-Link TL-WR1043ND berdasarkan parameter sumberdaya komputasi dan jaringan dalam arsitektur SDN?
3. Bagaimana pengaruh penggunaan Openflow terhadap kinerja router Linksys WRT54GL dan TP-Link TL-WR1043ND berdasarkan parameter sumberdaya komputasi dan jaringan?

## 1.3 Tujuan

Pada penelitian ini terdapat beberapa tujuan yang ingin dicapai oleh penulis, yaitu sebagai berikut.

1. Mengetahui kinerja Openflow-based switch pada router Linksys WRT54GL dan TP-Link TL-WR1043ND berdasarkan parameter sumberdaya komputasi dan jaringan pada arsitektur SDN.
2. Mengetahui pengaruh penggunaan Openflow terhadap kinerja router Linksys WRT54GL dan TP-Link TL-WR1043ND.

#### 1.4 Manfaat

Memberikan informasi terkait kinerja dan pengaruh penambahan protokol Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND sebagai SDN switch berdasarkan parameter penggunaan sumberdaya CPU dan memory, round trip time, throughput dan waktu pembentukan *flow table*. Juga dapat menjadi bahan pertimbangan/acuan dalam mengimplementasikan router sebagai SDN switch.

#### 1.5 Batasan Masalah

Untuk menghindari luasnya pembahasan, maka pada proposal ini penulis memberikan beberapa batasan permasalahan, yaitu :

1. Router yang digunakan adalah router Linksys WRT54GL dan router TP-Link TL-WR1043ND
2. Host yang digunakan berjumlah 5 buah dan berupa *virtual machine*
3. Topologi yang digunakan berbentuk linier
4. Menggunakan Openflow sebagai protokol SDN
5. Menggunakan Ryu sebagai *controller* pada SDN
6. Menggunakan algoritma *simple switch* pada *library* Ryu
7. Kinerja yang diuji berupa kecepatan waktu respon router ke *controller*, penggunaan *memory* dan CPU, *throughput* dan *round trip time*

#### 1.6 Sistematika Pembahasan

Sistematika pembahasan ditunjukkan untuk memberikan gambaran dan uraian dari penulisan skripsi ini secara garis besar yang meliputi beberapa bab sebagai berikut:

##### Bab I : Pendahuluan

Bab ini berisi mengenai latar belakang pembuatan laporan skripsi, rumusan masalah, batasan masalah, tujuan yang ingin dipenuhi, manfaat yang didapat dan sistematika penulisan pada penelitian ini

## **Bab II : Tinjauan Pustaka dan Dasar Teori**

Bab ini berisi tentang teori-teori pendukung serta kajian terhadap penelitian-penelitian terdahulu yang relevan. Teori dan kajian tersebut dapat menguatkan dan menjadi dasar penelitian maupun penulisan laporan

## **Bab III : Metodologi Penelitian**

Bab ini menguraikan dan membahas urutan dan langkah-langkah kerja pada penelitian ini. Bab ini terdiri atas studi literatur, analisis kebutuhan, perancangan sistem, pengujian, teknik analisis dan teknik pengambilan kesimpulan.

## **Bab IV : Perancangan dan Implementasi Lingkungan Pengujian**

Bab ini berisi tentang tahapan perancangan yang dilakukan untuk pengujian dan pengambilan data berdasarkan metodologi penelitian yang telah dibuat sebelumnya.

## **Bab V : Hasil dan Pembahasan**

Bab ini berisi tentang hasil yang telah didapat pada proses pengujian. Pada bab ini, data yang didapat dikumpulkan dan disajikan secara rapi dan sistematis dalam bentuk tabel ataupun grafik. Kemudian data tersebut dianalisis untuk didapatkan informasi mengenai kinerja router berdasarkan parameter yang digunakan.

## **Bab VI : Kesimpulan dan Saran**

Bab ini memuat kesimpulan yang diperoleh dari hasil pengujian dan analisis pada penelitian kinerja *Openflow-based switch* pada router Linksys WRT54GL dan TP-Link TL-WR1043ND dalam arsitektur *software defined networking*. Pada bab ini juga terdapat saran-saran terkait pengembangan penelitian selanjutnya, yang bertujuan untuk perbaikan-perbaikan dimasa yang akan datang.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Pada subbab ini dilakukan kajian terhadap, beberapa penelitian sebelumnya yang terkait dengan penelitian ini dan dijadikan sebagai pedoman dalam pelaksanaan penelitian. Berikut tabel penelitian sebelumnya dan saat ini

**Tabel 2.1 Kajian Pustaka**

No	Nama Penulis, Tahun dan Judul	Persamaan	Perbedaan	
			Penelitian Sebelumnya	Rencana Penelitian
1	Nurfani Abdillah [2016]. Analisis Kinerja Arsitektur <i>Software defined Network</i> dengan Openflow pada Mikrotik RB750	Mengimplementasikan Openflow pada router di arsitektur SDN	Menggunakan router Mikrotik RB750 Menggunakan <i>controller</i> Floodlight Analisis kinerja berupa <i>latency</i> , <i>throughput</i> dan <i>jitter</i>	Menggunakan router Linksys WRT54GL dan TP-Link TL-WR1043ND Menggunakan <i>controller</i> Ryu Analisis kinerja berupa penggunaan sumberdaya CPU dan memory, <i>round trip time</i> , <i>throughput</i> dan waktu pembentukan <i>flow table</i>
2	Rikie Kartadie dan Tommi Suryanto [2015]. Uji Performa <i>Software-Based Openflow Switch</i> Berbasis Openwrt	Mengimplementasikan Openflow pada router TP-Link TL-WR1043ND di arsitektur SDN	Menggunakan router TP-Link TL-WR1043ND dan Mininet Menggunakan <i>controller</i> floodlight Analisis kinerja berupa <i>latency</i> ,	Menggunakan router Linksys WRT54GL dan TP-Link TL-WR1043ND Menggunakan <i>controller</i> Ryu Analisis kinerja berupa penggunaan sumberdaya CPU

			troughput dan jitter	dan memory, <i>round trip time</i> , <i>throughput</i> dan waktu pembentukan <i>flow table</i>
3	Muhammad Hikam Hidayat [2017]. Implementasi Dan Analisis Kinerja Arsitektur Softwaredefined Network Berbasis Opendaylight Controller	Mengimplementasikan Openflow pada router di arsitektur SDN	Menggunakan router Mikrotik  Menggunakan <i>controller</i> opendaylight  Analisis kinerja berupa <i>latency</i> , <i>troughput</i> dan jitter	Menggunakan router Linksys WRT54GL dan TP-Link TL-WR1043ND  Menggunakan <i>controller</i> Ryu  Analisis kinerja berupa penggunaan sumberdaya CPU dan memory, <i>round trip time</i> , <i>throughput</i> dan waktu pembentukan <i>flow table</i>

Berdasarkan tabel 2.1, fokus penelitian ini adalah mendapatkan data kinerja Openflow pada Linksys WRT54GL dan TP-Link TL-TL-WR1043ND pada arsitektur SDN berdasarkan parameter sumberdaya komputasi dan jaringan.

## 2.2 Dasar Teori

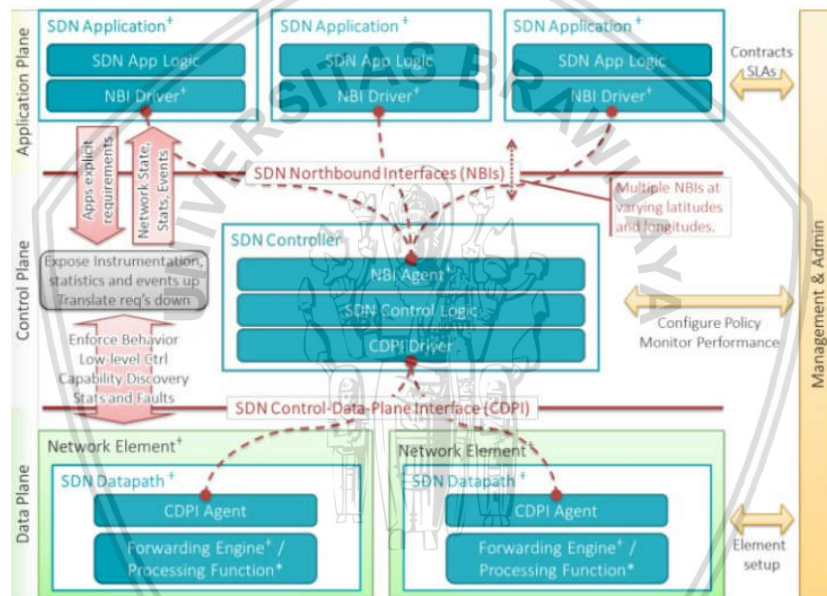
Pada subbab ini dibahas mengenai teori-teori yang mendukung dalam proses penelitian ini. Teori tersebut digunakan dalam penelitian analisis kinerja Openflow pada Router Linksy WRT54G dan TP-Link WR1043ND.

### 2.2.1 Software Defined Network

*Software defined Network* (SDN) merupakan sebuah gagasan yang membahas mengenai sebuah konsep baru dalam manajemen dan mengimplementasikan jaringan, terlebih untuk mempermudah pengelolaan jaringan yang kompleks (mulyana, 2014). Prinsip dasar SDN adalah dengan memisahkan antara *control* dan forwarding *plane*.

- 1 *Standard interface* untuk memprogram perangkat jaringan
- 2 *Control-plane* yang terpusat atau adanya pengendali pada jaringan yang mampu membentuk peta logika dari seluruh jaringan. Peta tersebut kemudian direpresentasikannya melalui API
- 3 *Virtualisasi* dimana beberapa sistem operasi jaringan dapat mengontrol bagian-bagian (slices atau substrates) dari perangkat yang sama.

Dalam konsep SDN, tersedia open *interface* yg memungkinkan sebuah entitas software/aplikasi untuk mengendalikan konektivitas yg disediakan oleh sejumlah sumber-daya jaringan, mengendalikan aliran trafik yg melewatinya serta melakukan inspeksi terhadap atau memodifikasi trafik tersebut. Gambar berikut menunjukan arsitektur SDN beserta komponen dan interaksinya.



**Gambar 2.1 Komponen SDN**

Sumber: (*Open network fondation*, 2013)

Berdasarkan gambar 2.1 diatas, arsitektur SDN dapat dilihat sebagai 3 lapis/bidang:

- 1 infrastruktur (*data-plane / infrastructure layer*): terdiri dari elemen jaringan yg dapat mengatur SDN Datapath sesuai dengan instruksi yg diberikan melalui *Control-Data-Plane Interface (CDPI)*
- 2 kontrol (*control plane / layer*): entitas kontrol (*SDN Controller*) mentranslasikan kebutuhan aplikasi dengan infrastruktur dengan memberikan instruksi yg sesuai untuk SDN Datapath serta memberikan informasi yg relevan dan dibutuhkan oleh *SDN Application*

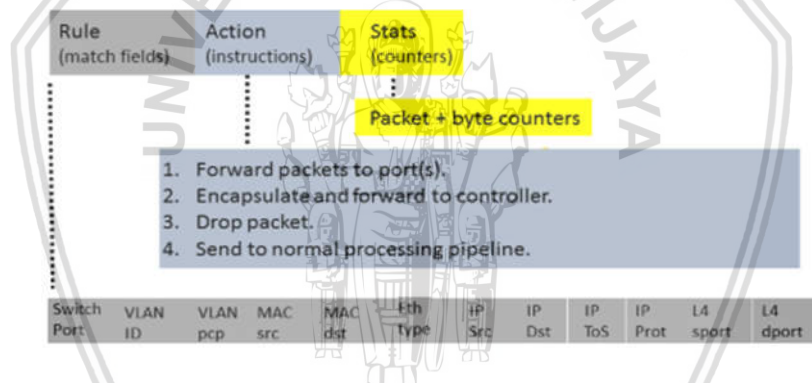


- 3 aplikasi (*application plane / layer*): berada pada lapis teratas, berkomunikasi dengan sistem via *NorthBound Interface* (NBI)

### 2.2.2 Openflow

Openflow adalah protokol yang dikembangkan *open networking foundation* untuk menangani proses komunikasi antara *control plane* dan *data plane*. Openflow merupakan standar komunikasi protokol pertama pada arsitektur SDN. Openflow memungkinkan *controller* untuk mengirimkan perintah kepada *switch* terkait paket yang ada pada *switch*. Openflow mendukung berbagai jenis vendor dalam sebuah jaringan komputer.

Openflow bekerja dengan cara mengecek entri pada *flow table*, jika tidak memiliki entri yang sesuai, *switch* akan mengirimkan paket ke *controller*. *Controller* selanjutnya mengambil keputusan pada paket tersebut. *controller* bisa melakukan drop atau menambahkan *flow* tentang cara meneruskan paket yang sejenis kedepannya. Setiap *flow* tabel terdiri atas *flow entries* dan setiap *flow* entries tersusun atas 3 komponen seperti pada gambar 2.2 berikut.



**Gambar 2.2 Flow entries pada Openflow**

Sumber : (James F. Kurose, 2014)

Berikut penjelasan dari gambar *flow table* diatas.

1. *Match field*: digunakan untuk menentukan kondisi paket yang sesuai menuju *flow* yang tepat.
2. *Counter*: digunakan untuk menghitung matching paket yang telah terjadi untuk tujuan manajemen.
3. *Instruction*: menentukan tindakan yang akan diterapkan ke *flow* tertentu.

### 2.2.3 Ryu

Ryu merupakan sebuah perangkat lunak yang digunakan sebagai *controller* pada arsitektur jaringan SDN. Sebuah *controller* pada arsitektur SDN berperan sebagai otak pada sebuah jaringan yang mengomunikasikan informasi antar router ataupun *switch*. (SDXcentral, 2014). Penggunaan Ryu pada arsitektur SDN dapat mempermudah dalam mengatur, manajemen jaringan dan mengontrol aplikasi yang ada didalamnya. Bahasa pemrograman yang didukung Ryu adalah Bahasa python dan dapat menggunakan protokol Openflow, NetConf, Of-config dll

### 2.2.4 Router

Router adalah perangkat jaringan yang menghubungkan beberapa perangkat pada jaringan yang memiliki subnet yang berbeda. Subnet merupakan jaringan yang terdiri dari satu set *host* yang memiliki prefix yang sama. Router terdiri dari processor, *memory* dan input/output interface. Pada *memory* router terdapat sebuah sistem operasi yang berbeda-beda pada setiap jenis dan vendornya. Sistem operasi router berbeda dengan sistem operasi komputer pada umumnya, ukuran sistem operasi router jauh lebih kecil dan mempunyai fitur yang sangat terbatas. Beberapa contoh sistem operasi router antara lain Openwrt, DD-WRT, IOS dll. Sistem operasi ini dibuat menjadi binary *firmware* image atau biasa disebut *firmware* router.

#### 2.2.4.1 Router Linksys WRT54GL

Linksys WRT54GL merupakan Wireless Broadband Router yang didalamnya terdapat 3 fungsi dasar yaitu sebagai Access Point (54Mbps Wireless-G / 802.11g dan 11Mbps Wireless-B / 802.11b), sebagai *Switch* yang terdiri dari 4 *port* full duplex 10/100 Ethernet dan sebagai Router yang berfungsi untuk manage dan sharing koneksi internet. Berikut fitur-fitur bawaan Linksys WRT54GL:



**Gambar 2.3 Router Linksys WRT54GL**

Sumber: (www.Linksys.com, 2017)

Berdasarkan gambar 2.3 diatas, router ini memiliki dua konektor antenna RP-TNC eksternal yang berfungsi untuk meningkatkan kinerja access point, 1 *port* koneksi kabel ke internet (WAN), 4 Port ethernet switch dan sebuah tombol reset. Untuk mengkonfigurasi Salah satu keunggulan dari Linksys WRT54GL ini Linksys WRT54GL kita bisa menggunakan web browser seperti Chrome dengan alamat *default* 192.168.1.1 dengan user admin dan password kosong. untuk adalah bisa upgrade *firmware*-nya dengan DD-WRT sehingga menambah kemampuan dari wireless router tersebut dari fungsi bawaannya yang sebatas access point menjadi sebuah WDS / Repeater dan juga tambahan fitur-fitur yang lain seperti bandwidth management.

#### 2.2.4.2 Router TP-Link TL-WR1043ND

TP-Link TL-WR1043ND merupakan router Gigabit yang mengintegrasikan *Switch 4-port*, Firewall, NAT-Router dan AP nirkabel Wireless N Gigabit Router memberikan jangkauan dan kecepatan yang luar biasa, yang bisa sepenuhnya memenuhi kebutuhan jaringan Kantor Kecil / Rumah Tangga



TL-WR1043ND

**Gambar 2.4 Router TP-Link TL-WR1043ND**

Sumber: (www.TP-Link.com, 2017)

Berdasarkan gambar 2.4 diatas, terdapat 2 buah *interface* pada router TP-Link TL-WR1043ND yaitu, eth0 dan eth1. Pada eth0 *interface* lan dan wireless digabung, sedangkan pada *interface* eth1 terdapat wan. Berikut fitur-fitur yang ada pada TP-Link TL-WR1043ND.

1. Sesuai dengan IEEE 802.11n untuk menyediakan data nirkabel hingga 300Mbps.
2. Berbagi data dan akses internet bagi pengguna, mendukung Dynamic ip / Static ip / PPPoE Internet
3. mendukung *Virtual Server*, Aplikasi Khusus dan *host* DMZ.
4. Mendukung *Flow Statistics*.

## 5. Mendukung upgrade *firmware* dan manajemen Web.

Untuk melakukan upgrade firmware pada router TP-Link TL-WR104ND, dapat dilakukan dengan menggunakan browser seperti firefox atau chrome dengan alamat default 192.168.1.1 atau alamat yang tertera pada buku petunjuk router.

### 2.2.5 Openwrt

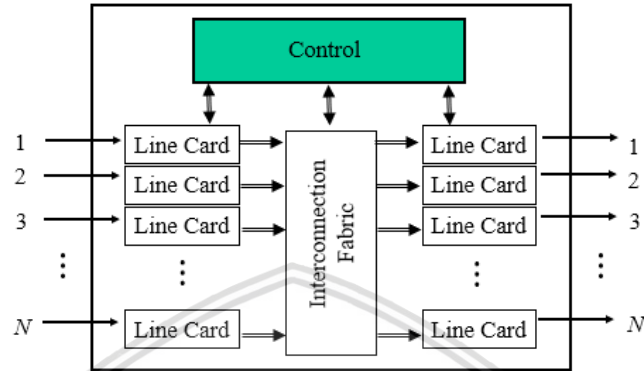
Openwrt merupakan sebuah sistem operasi berbasis kernel linux yang biasa digunakan pada perangkat jaringan. komponen utama dari Openwrt meliputi kernel linux, util linux, uclib ataumusl dan bussybox. Semua kompoonen telah dioptimalkan sesuai dengan ukuran perangkat yang kecil sehingga akan pas pada penyimpanan *memory* sebuah router rumahan. Openwrt dikonfigurasi menggunakan antarmuka baris perintah (ash shell), atau antarmuka web (LuCI). Ada sekitar 3500 paket perangkat lunak opsional yang tersedia untuk instalasi melalui sistem pengelolaan paket opkg. Openwrt dapat berjalan pada berbagai jenis perangkat, termasuk router CPE, residentia Gateway, smartphone, pocket komputer (misalnya Ben NanoNote), dan laptop. Hal ini juga memungkinkan untuk menjalankan Openwrt pada komputer pribadi, yang paling sering didasarkan pada x86 arsitektur. berikut beberapa fitur yang ada pada Openwrt:

- 1 Dapat ditulisi *filesystem* root, memungkinkan pengguna untuk menambahkan, menghapus atau memodifikasi setiap *file*. Hal ini dicapai dengan menggunakan overlayfs untuk overlay hanya-baca dikompresi SquashFS sistem *file* dengan JFFS2 *file* system dapat ditulisi dalam mode salinan pada menulis. JFFS2 mendukung flash wear leveling.
- 2 Opkg manajer paket, yang mirip dengan dpkg, memungkinkan pengguna untuk menginstal dan menghapus perangkat lunak. Paket repositori berisi sekitar 3500 paket. Hal ini kontras dengan *firmwares* berbasis Linux yang berdasarkan sistem hanya-baca *file* tanpa kemungkinan untuk memodifikasi piranti lunak yang dipasang tanpa membangun kembali dan berkedip gambaran lengkap *firmware*.
- 3 Serangkaian skrip yang disebut UCI (bersatu konfigurasi antarmuka) dimaksudkan untuk menyatukan dan menyederhanakan konfigurasi sistem keseluruhan
- 4 Extensible konfigurasi dan driver hardware seluruh, misalnya built-in jaringan *switch* VLAN-kemampuan mereka WNICs, DSL modem, FX, tombol tersedia hardware, dll

### 2.2.6 Arsitektur *Switch*

*Switch* merupakan perangkat jaringan yang terdiri dari beberapa port dan berfungsi untuk menerima informasi pada salah satu portnya, kemudian

mengirimkan informasi tersebut ke satu atau beberapa port lainnya yang telah ditentukan. Secara umum *switch* terdiri dari *input/output interface*, *interconnection fabric* dan *controller* seperti yang terlihat pada gambar 2.5 berikut.



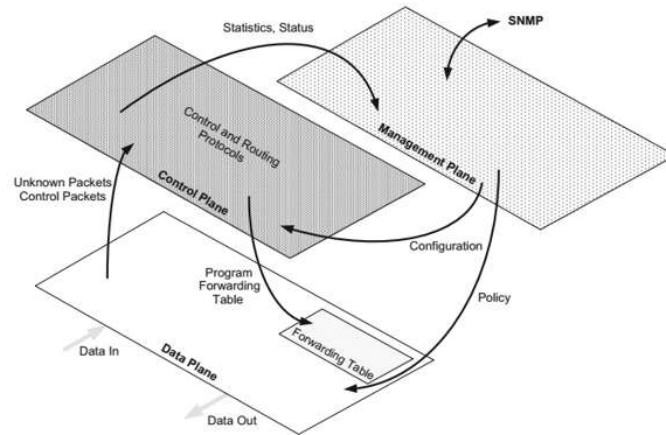
**Gambar 2.5 Struktur paket switch/router**

Sumber: (James F. Kurose, 2013)

#### 2.2.6.1 *Reguler Switch*

Pada *Reguler switch* komunikasi *switch* dibagi menjadi 2 yaitu horizontal view dan vertikal view. Horizontal view merupakan hubungan antar perangkat dalam sebuah topologi. Sedangkan vertikal view, merupakan komunikasi antar *plane* dalam sebuah perangkat. *Plane* pada *reguler switch* dibagi menjadi 3 bagian seperti yang terlihat pada gambar 2.6 dibawah, berikut penjelasan dari masing-masing *plane*.

1. *Data plane*, sebagai paket *buffering*, paket *scheduling*, *forwarding* dan menangani penerimaan paket melalui input dan *output port*.
2. *Controll plane*, mengelolah informasi pada *forward table* dan memproses beberapa protokol kontrol yang mempengaruhi *forwarding table*.
3. *Management plane*, tempat network administrator mengkonfigurasi dan memonitor *switch*.



**Gambar 2.6 Plane pada switch tradisional**

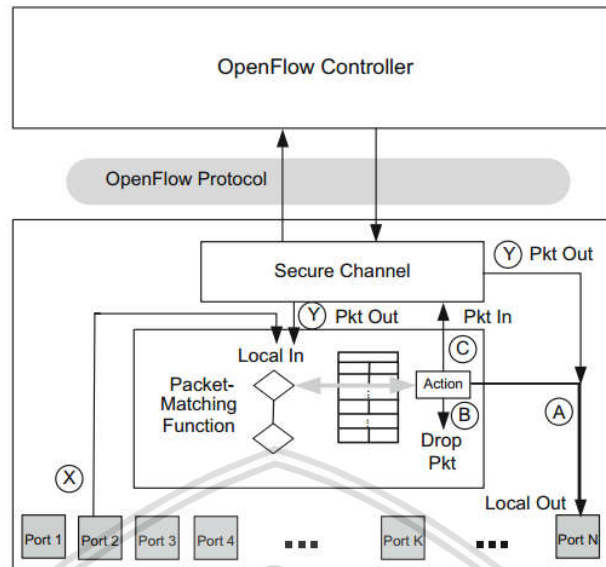
Sumber: (Goransson, 2014)

### 2.2.6.2 SDN Switch

Pada arsitektur SDN, *switch* hanya berperan sebagai *data plane*. Untuk dapat bekerja sebagai *SDN switch*, diperlukan sebuah protokol SDN seperti Openflow. Openflow *switch* bekerja dengan menangani paket sesuai intruksi yang ada pada *controller*. Aturan Openflow terdiri dari 3 aksi yang dapat dilakukan *SDN switch* berdasarkan instruksi dari *controller* yaitu sebagai berikut.

1. *Forward*  
Paket akan diforward apabila sesuai dengan instruksi *controller* pada *flow* tabel
2. *Drop packet*  
Paket akan dibuang apabila tidak sesuai dengan instruksi *controller* pada *flow* tabel
3. *Pass to controller*  
Apabila informasi paket mengharuskan diproses pada *controller* maka, paket akan diforward ke *controller*

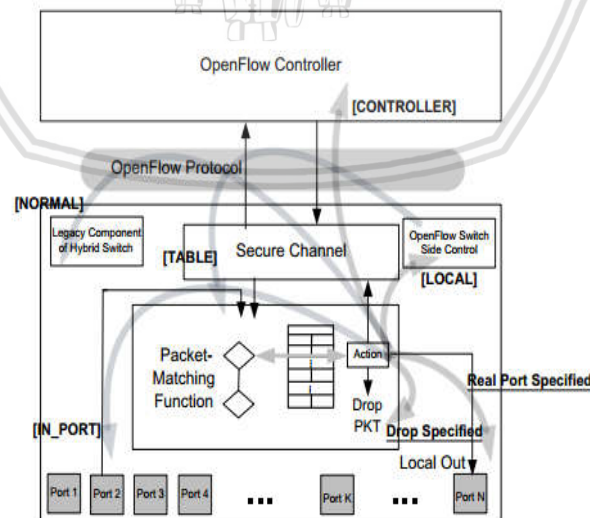




**Gambar 2.7 Openflow switch**

Sumber: (Goransson, 2014)

Terdapat 2 jenis Openflow switch yaitu Openflow-only switch dan Openflow-hybrid switch. Openflow-only switch merupakan switch yang telah terintegrasi dengan Openflow sejak awal dan hanya memproses paket dengan aturan Openflow. Openflow-hybrid terdiri dari 2 mode yaitu Openflow mode dan legacy mode. Pada Openflow mode switch mengikuti aturan Openflow sedangkan pada mode legacy switch bekerja seperti switch/router tradisional. Pada Openflow-hybrid terdapat proses preprocessing classification untuk menentukan sebuah paket diproses melalui proses Openflow atau traditional switching.



**Gambar 2.8 Jalur paket Openflow switch**

Sumber: (Goransson, 2014)

### 2.2.7 Latency

*Latency* merupakan banyaknya waktu yang dibutuhkan untuk mengirim pesan dari sumber ke tujuan atau bisa disebut pengiriman satu kali jalan. Berhubungan dengan *latency*, terdapat istilah *round trip time*. *Round trip time* merupakan waktu yang dibutuhkan untuk mengirim pesan dari sumber ke tujuan dan kembali ke sumber lagi atau bisa disebut pengiriman 2 kali jalan. *Latency* dan *round trip time* diukur dalam bentuk waktu dan biasanya menggunakan satuan miliseconds (ms). Terkait *latency* terdapat istilah round-trip time atau *round trip time*, yaitu seberapa lama waktu yang dibutuhkan untuk mengirim pesan dari ujung A ke B dan sebaliknya (Syafrizal, 2015). Besarnya *latency* pada suatu jaringan dipengaruhi oleh 3 komponen yaitu sebagai berikut.

1. *Speed-of-light propagation delay*

*Speed of light propagation delay* terjadi karena tidak ada satupun media yang dapat memiliki kecepatan melebihi kecepatan cahaya, termasuk kabel jaringan. Jika kita mengetahui jarak dari dua titik, kita dapat menghitung *latency* dari kecepatan cahaya tetapi kita harus berhati-hati karena cahaya dapat merambat melalui medium yang berbeda dengan kecepatan yang berbeda pula. Di ruang hampa kecepatan cahaya memiliki kecepatan  $3.0 \times 10^8$  m/s. Jika melalui kabel kecepatannya menurun menjadi  $2.3 \times 10^8$  m/s, sedangkan di fiber kecepatannya menjadi  $2.0 \times 10^8$  m/s.

2. Waktu untuk transmisi data (Transmit)

waktu untuk melakukan transmisi suatu data yang merupakan bagian dari bandwidth jaringan dengan mempertimbangkan ukuran (size) dari paket yang dibawa.

3. Antrian dalam Jaringan (Queue)

Antrian yang mungkin terjadi di dalam jaringan misalnya paket yang disimpan beberapa saat sebelum di forward ke beberapa lokasi.

Berdasarkan ketiga komponen diatas didapat persamaan pada *latency* sebagai berikut.

1.  $Latency = Propagation + Transmit + Queue$
2.  $Propagasi = jarak / Speed\ of\ Light$
3.  $Transmit = Size / Bandwidth$

### 2.2.8 Throughput

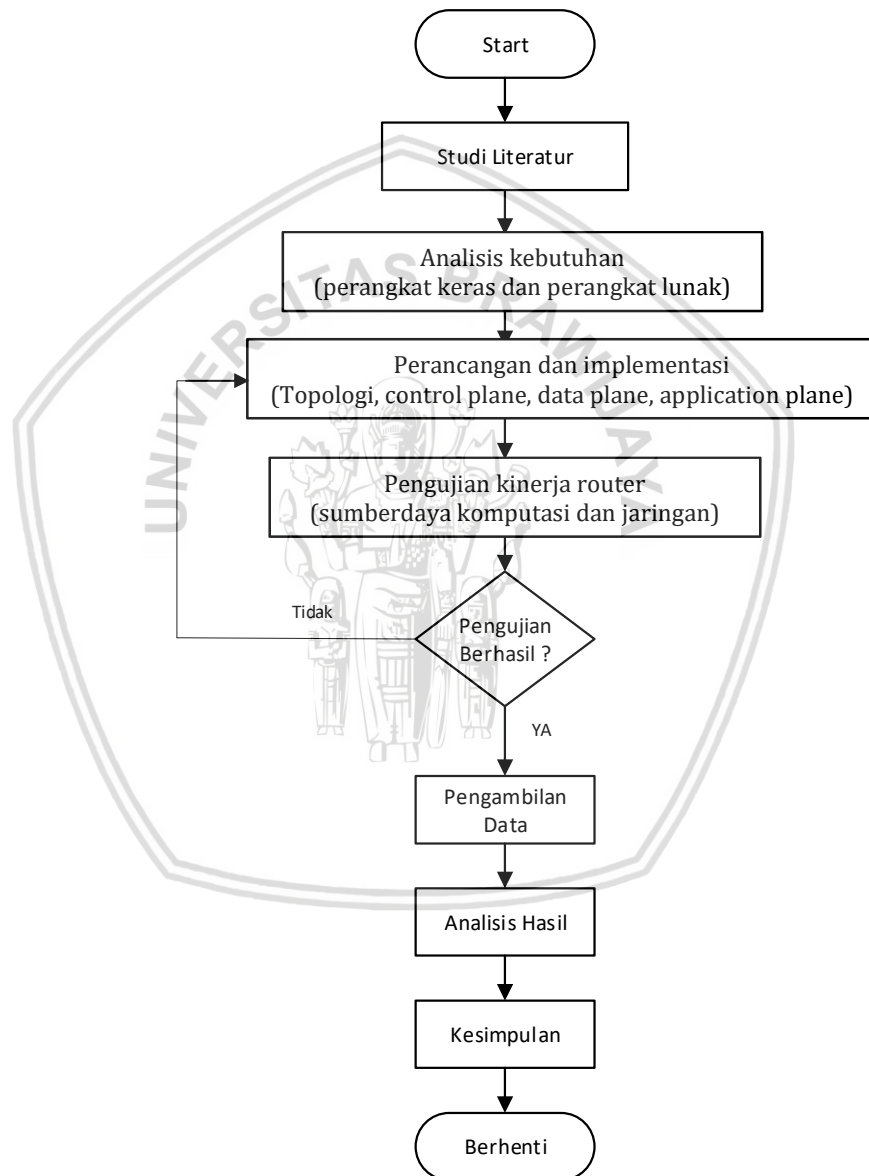
*Throughput* merupakan kecepatan informasi yang datang pada waktu tertentu. Throughput memiliki kaitan erat dengan *bandwidth* atau kemampuan maksimum untuk mengalirkan data/informasi dalam satuan waktu tertentu, seperti detik. Throughput lebih mengarah pada *bandwidth* aktual atau *bandwidth* pada waktu tertentu, pada kondisi dan jaringan internet tertentu. Salah satu contoh throughput dapat dilihat ketika melakukan transfer data dengan ukuran tertentu, kecepatan transfer tersebut merupakan *throughput* (Syafriзал, 2015). Terdapat beberapa faktor yang mempengaruhi *throughput* yaitu sebagai berikut.

1. Spesifikasi perangkat jaringan.
2. Tipe data yang ditransfer.
3. Topologi jaringan.
4. Banyaknya pengguna jaringan.
5. Spesifikasi komputer client/user.
6. Spesifikasi komputer server.



### BAB 3 METODOLOGI

Pada bab ini dijelaskan langkah-langkah yang akan dilakukan dalam pengerjaan skripsi, yaitu studi literatur, analisis kebutuhan, perancangan dan implementasi lingkungan pengujian, pengujian dan pengambilan data, analisis hingga kesimpulan. Berikut merupakan diagram alir urutan pengerjaan penelitian ini yang dapat dilihat pada gambar 3.1.



Gambar 3.1 *Flowchart* pengerjaan penelitian

### 3.1 Studi Literatur

Sebagai dasar dan landasan mengenai penelitian, dilakukan proses studi literatur. Studi literatur tersebut meliputi dasar-dasar SDN, Openwrt, router Linksys WRT54GL dan TP-Link TL-WR1043ND, penerapan Openflow pada router Openwrt serta penggunaan *controller* Ryu pada arsitektur SDN. Studi literatur mengenai penelitian sebelumnya terkait implementasi SDN pada router ataupun penerapan SDN menggunakan *emulator* juga dibutuhkan agar proses penelitian berjalan lebih baik berdasarkan penelitian sebelumnya tersebut.

### 3.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan oleh sistem yang akan diuji. Analisis kebutuhan dilakukan dengan mengidentifikasi kebutuhan sistem dan siapa saja yang terlibat di dalamnya. Untuk mempermudah penelitian ini, kebutuhan di kelompokkan menjadi 2 jenis yaitu kebutuhan fungsional dan kebutuhan non-fungsional

#### 3.2.1 Kebutuhan Fungsional

kebutuhan fungsional merupakan kebutuhan akan hal-hal yang dapat dilakukan sistem dan keluaran atau informasi apa saja yang harus dihasilkan pada sistem tersebut. Kebutuhan fungsional ini dikelompokkan menjadi 3 bagian, yaitu sebagai berikut.

##### 3.2.1.1 Router

- Menjalankan protokol Openflow
- Membuat *forwarding table* Openflow
- *Forwarding* paket antar *host*

##### 3.2.1.2 Controller

- Menjalankan aplikasi Ryu
- Menjalankan aplikasi *simple\_switch* pada Ryu

##### 3.2.1.3 Host

- Mengirim dan menerima *file* antar *host*

#### 3.2.2 Kebutuhan Non Fungsional

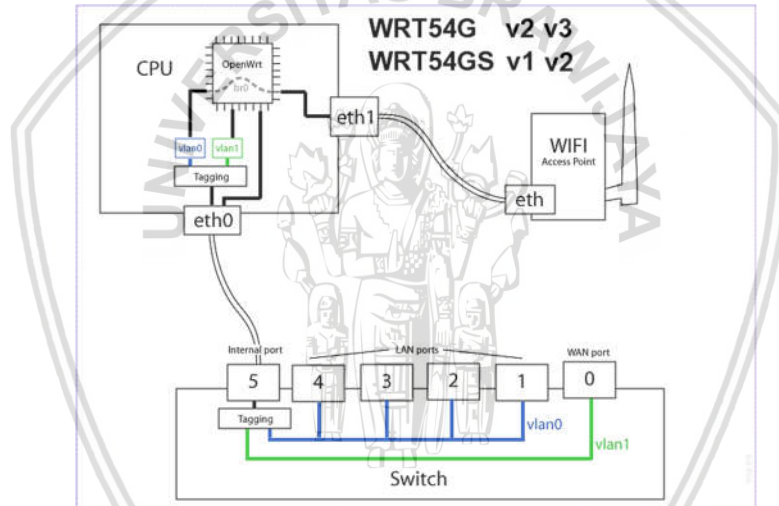
Kebutuhan non-fungsional merupakan kebutuhan yang terkait dengan komponen yang ada pada sistem. Komponen tersebut meliputi perangkat keras dan perangkat lunak

### 3.2.2.1 Perangkat Keras

Perangkat keras yang dibutuhkan dalam penelitian ini adalah sebagai berikut:

1 1 Router Linksys WRT54GL dengan spesifikasi:

- 200 MHZ CPU Broadcom BCM5352
- 16 MB RAM
- 4 MB *flash memory*
- 4 fast ethernet *port*
- 1 wan *port*
- 802.11b/gn *wireless support*



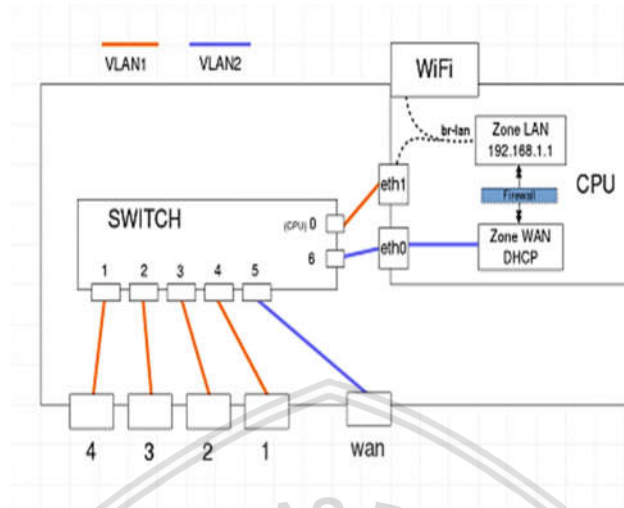
Gambar 3.2 Blok diagram Linksys WRT54GL

Sumber: (wiki.Openwrt.org, 2013)

2 1 router TP-Link TL-WR1043ND dengan spesifikasi

- 400 MHZ CPU Atheros AR9132
- 32 MB RAM
- 8 MB *flash memory*
- 4 gigabit *ethernet port*
- 1 wan *port*

- 802.11b/gn wireless support



**Gambar 3.3 Blok diagram TP-Link TL-WR1043ND**

Sumber: (wiki.Openwrt.org, 2013)

- 3 2 Laptop sebagai *controller* dan *host*  
spesifikasi *controller* pada laptop 1.

- 2.00 GHZ CPU AMD A8-6410
- 4 GB RAM
- 1 TB *hardisk*

Spesifikasi laptop 2

- 3.00 GHZ CPU AMD A9-9420
- 4 GB RAM
- 500 GB *hardisk*

Spesifikasi *virtual host* pada laptop 1 dan 2

- 2.00 GHZ CPU AMD A8-6410 dan 3.00 GHZ CPU AMD A9-9420
- 512 RAM
- 20 GB Hardisk



- 4 2 buah kabel ethernet

### 3.2.2.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam penelitian ini meliputi compiler dan aplikasi untuk merancang perangkat keras. Perangkat lunak yang dibutuhkan yaitu:

- 1 Ubuntu 16.04 64bit *desktop* sebagai sistem operasi pada *controller*
- 2 Ubuntu 15.04 64bit *server* sebagai sistem operasi pada *host*
- 3 Openwrt 10.3 sebagai sistem operasi pada router
- 4 Openflow sebagai protokol standard SDN yang diinstal pada router
- 5 Ryu sebagai software *controller*
- 6 Aplikasi *simple\_switch* pada *library* Ryu sebagai *application plane*
- 7 Aplikasi iftop, top, ping untuk melakukan pengujian

## 3.3 Perancangan dan Implementasi Lingkungan Pengujian

Perancangan dan implementasi lingkungan pengujian merupakan tahapan yang berisi tentang perencanaan proses penelitian dan implementasi skenario lingkungan pengujian pada perangkat yang digunakan.

### 3.3.1 Perancangan

Dalam perancangan dijelaskan mengenai perencanaan tahapan-tahapan dalam mengimplementasikan lingkungan pengujian dan perancangan terhadap perangkat yang akan digunakan pada proses pengujian. Tahapan perancangan dibagi menjadi 4 bagian yaitu perancangan topologi, *data plane*, *control plane* dan *application plane*

### 3.3.2 Implementasi lingkungan pengujian

Implementasi lingkungan pengujian merupakan realisasi dari perencanaan yang telah dibuat pada tahapan perancangan. Pada tahap ini dilakukan proses instalasi dan konfigurasi pada perangkat yang digunakan. Proses instalasi dan konfigurasi meliputi instalasi *firmware*, instalasi aplikasi *controller* dan konfigurasi router.



### 3.3.3 Pengujian

Dalam tahap ini dijelaskan bagaimana pengujian kinerja Openflow-based switch pada router Linksys WRT54GL dan TP-Link TL-WR1043ND. Penjelasan tersebut meliputi skenario pengujian dan proses pengambilan data pada masing-masing parameter. Skenario pengujian pada setiap parameter dikelompokkan berdasarkan jenis router dan variabel penguji. Variabel penguji pada setiap parameter mempunyai jenis yang berbeda-beda. Pada skenario pengujian penggunaan CPU dan memory terdapat 2 variabel yang digunakan yaitu ukuran *file* dan jumlah *host*. Pada skenario pengujian roundtrip time, besar paket dijadikan sebuah variabel penguji. Sedangkan parameter *throughput* dan waktu pembentukan *flow table* tidak digunakan variabel penguji.

### 3.4 Hasil dan Pembahasan

Hasil dan pembahasan dilakukan setelah proses pengujian telah berhasil. Pada bab ini dijelaskan tentang penjabaran data dan analisis kinerja pada masing-masing parameter yang telah didapat pada proses pengujian. Hasil dan pembahasan tersebut antara lain, waktu pembentukan *flow table*, penggunaan *memory* dan CPU, *Round trip time* dan *throughput*.

### 3.5 Kesimpulan dan Saran

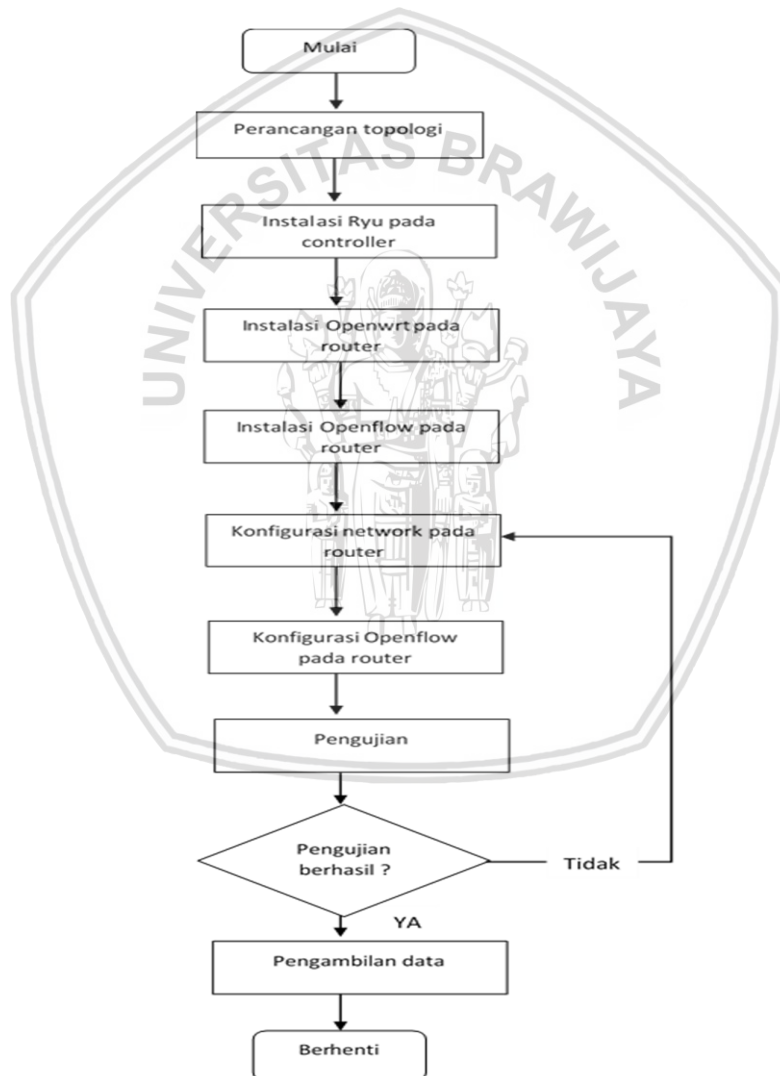
Dalam bab ini dijelaskan tentang kesimpulan yang dirangkum dari bab-bab sebelumnya beserta saran-saran untuk pengembangan penelitian kedepannya. Pengambilan kesimpulan dilakukan berdasarkan data hasil pengujian dan analisis kinerja Openflow menggunakan parameter sumberdaya komputasi dan jaringan.

## BAB 4 PERANCANGAN DAN IMPLEMENTASI LINGKUNGAN PENGUJIAN

### 4.1 Perancangan

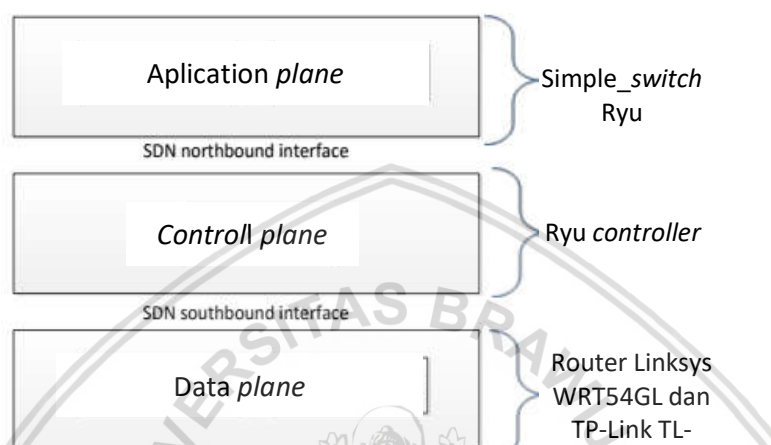
#### 4.1.1 Perancangan Lingkungan pengujian

Sebelum dilakukan implementasi lingkungan pengujian, perlu dilakukan perancangan lingkungan pengujian agar proses implementasi tersebut dapat dilakukan secara runtut dan efektif. Berikut perancangan lingkungan pengujian dapat dilihat pada gambar 4.1



Gambar 4.1 Skema alur perancangan lingkungan pengujian

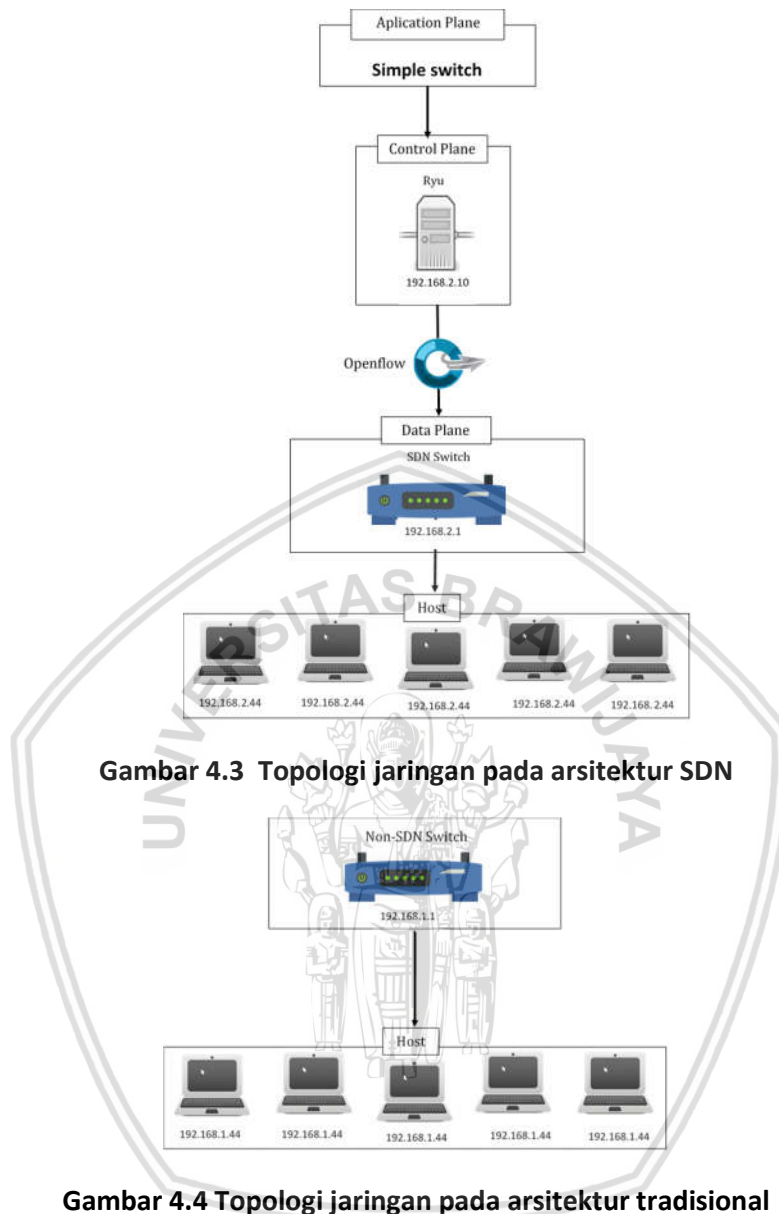
Berdasarkan gambar 4.1, tahapan perancangan lingkungan pengujian dapat dikelompokkan menjadi 4 bagian. Tahapan tersebut meliputi perancangan topologi, data *plane*, *control plane* dan *application plane*. Pada layer data *plane* terdapat router Linksys wrt54gl dan TP-Link tl-TL-WR1043ND. Pada layer *control plane* digunakan laptop yang diinstal aplikasi Ryu. Pada *application plane* digunakan aplikasi *simple\_switch* yang diambil dari *library* Ryu. Penggunaan ketiga layer tersebut dapat dilihat pada gambar 4.2



**Gambar 4.2 Penggunaan layer pada SDN**  
Sumber: (www.sdxcentral.com, 2015)

#### 4.1.2 Topologi

Pada penelitian ini router difungsikan sebagai switch, sehingga alamat jaringan pada setiap host berada pada *network* yang sama. Dengan memfungsikan router sebagai switch, pengujian dapat dilakukan sebanding antara arsitektur tradisional dan SDN yang menggunakan aplikasi *simple switch*. Topologi yang digunakan terdiri atas 5 *virtual host*, 1 *controller* dan 1 router pada arsitektur SDN. Pada arsitektur tradisional Topologi yang digunakan terdiri atas 5 *virtual host* dan 1 router. Perancangan topologi dibuat secara linier yaitu dengan cara menghubungkan ke 5 *host* secara langsung pada router. Bentuk topologi tersebut dipilih karena kelima *host* memiliki posisi yang sama pada jaringan dan router menduduki posisi *central*. Dengan topologi tersebut kinerja dari jaringan bergantung banyak pada kinerja router. Berikut topologi yang digunakan dapat dilihat pada gambar 4.3 dan 4.4



Gambar 4.4 Topologi jaringan pada arsitektur tradisional

## 4.2 Implementasi Lingkungan Pengujian

### 4.2.1 Control plane

Perancangan pada *control plane* meliputi instalasi dan konfigurasi aplikasi ryu pada laptop dengan system operasi ubuntu

#### 4.2.1.1 Instalasi Controller Ryu

Berikut ini langkah-langkah instalasi aplikasi Ryu sebagai *controller* pada SDN melalui source code pada github.

- 1 Menginstal aplikasi git yang digunakan untuk menyalin source code dari gith-hub. Instalasi dilakukan menggunakan terminal dengan perintah:

“apt-get instal git”

- 2 salin source code Ryu pada github menggunakan perintah:

“git clone git://github.com/osrg/Ryu.git”

- 3 Instal *file* setup pada folder Ryu yang di salin tadi menggunakan perintah:

“./setup.py instal”.

Setelah proses instalasi selesai, uji coba aplikasi Ryu apakah sudah benar terinstal atau belum dengan perintah: “Ryu-manager”.

#### 4.2.2 Data plane

Tahapan perancangan pada data *plane* meliputi instalasi *firmware* openwrt, instalasi Openflow dan konfigurasi router

##### 4.2.2.1 Instalasi Openwrt

Agar router bisa dijadikan sebagai SDN *switch* sistem operasinya harus diubah menjadi Openwrt. Instalasi dilakukan karena sistem operasi sebelumnya tidak mendukung protokol Openflow. Berikut langkah-langkah instalasi Openwrt pada router.

1. Download image Openwrt pada situs resminya yaitu <https://downloads.openwrt.org>. Openwrt yang digunakan adalah Openwrt versi 10.3 (backfire) dengan seri brcm47xx untuk Linksys dan ar71xx untuk TP-Link
2. Masuk konfigurasi pada web *interface* dengan mengakses alamat *default* router yaitu 192.168.1, login menggunakan username “admin” dan password “admin”. Masuk pada bagian *firmware* upgrade pada tab system tool untuk TP-Link dan tab administrator untuk Linksys. Setelah itu masukan image Openwrt yang sesuai pada masing-masing router kemudian klik upgrade, tunggu hingga proses upgrade selesai. Setelah
3. upgrade sukses router akan melakukan reboot dan sistem Openwrt telah siap digunakan.



- 3 Edit konfigurasi *network* router untuk mengatur penggunaan *virtual interface* dengan perintah:

`"vi /etc/config/network"`

Kemudian edit konfigurasi tersebut sesuai tabel 3.1 dan 3.2

**Tabel 4.1 Konfigurasi *network* Linksys WRT54GL**

<pre>#### Konfigurasi VLAN config switch eth0 config switch_vlan eth0_0     option device "eth0"     option vlan 0     option ports "0 5"  config switch_vlan eth0_1     option device "eth0"     option vlan 1     option ports "1 5"  config switch_vlan eth0_2     option device "eth0"     option vlan 2     option ports "2 5"  config switch_vlan eth0_3     option device "eth0"     option vlan 3     option ports "3 5"</pre>	<p>Pengaturan virtual lan(VLAN) yang digunakan pada router Linksys WRT54GL. Dalam hal ini digunakan 4 VLAN pada 4 port, sehingga vlan pada setiap port berbeda satu sama lain.</p>
<pre>#### Konfigurasi Interface  config interface     option ifname "eth0.3"     option proto static  config interface     option ifname "eth0.2"     option proto static  config interface     option ifname "eth0.1"     option proto static  config interface Lan     option type bridge     option ifname "eth0.0"     option proto static     option ipaddr 192.168.1.1     option netmask 255.255.255.0  config interface loopback     option ifname "lo"</pre>	<p>Pengaturan interface in/out yang digunakan pada router Linksys WRT54GL. Dalam hal ini terdapat 4 buah interface dengan Vlan yang berbeda pada setiap interfacenya. Setiap interface yang ada, menggunakan mode pengalamatan static dan eth0.0 sebagai bridge.</p>



```
option proto      static
option ipaddr     127.0.0.1
option netmask    255.0.0.0
```

**Tabel 4.2 Konfigurasi network TP-Link TL-WR1043ND**

#### Konfigurasi VLAN

```
config 'switch'
  option 'name' 'rtl8366rb'
  option 'enable_vlan' '1'
  option 'enable_learning' '0'
```

```
config 'switch_vlan'
  option 'device' 'rtl8366rb'
  option 'vlan' '1'
  option 'ports' '1 5t'
```

```
config 'switch_vlan'
  option 'device' 'rtl8366rb'
  option 'vlan' '2'
  option 'ports' '2 5t'
```

```
config 'switch_vlan'
  option 'device' 'rtl8366rb'
  option 'vlan' '3'
  option 'ports' '3 5t'
```

```
config 'switch_vlan'
  option 'device' 'rtl8366rb'
  option 'vlan' '4'
  option 'ports' '4 5t'
```

Pengaturan virtual lan(VLAN) yang digunakan pada router TP-Link TL-WR1043ND. Dalam hal ini digunakan 4 VLAN pada 4 port, sehingga vlan pada setiap port berbeda satu sama lain.

#### Konfigurasi Interface

```
config 'interface'
  option 'ifname' 'eth0.4'
  option 'proto' 'static'
```

```
config 'interface'
  option 'ifname' 'eth0.3'
  option 'proto' 'static'
```

```
config 'interface'
  option 'type' 'bridge'
  option 'ifname' 'eth0.2'
  option 'proto' 'static'
```

```
config 'interface Lan'

  option 'ifname' 'eth0.1'
  option 'proto' 'static'
  option 'ipaddr' '192.168.1.1'
  option 'netmask' '255.255.255.0'
```

Pengaturan interface in/out yang digunakan pada router TP-Link TL-WR1043ND. Dalam hal ini terdapat 4 buah interface dengan Vlan yang berbeda pada setiap interfacenya. Setiap interface yang ada, menggunakan mode pengalamatan static dan eth0.1 sebagai bridge.

```
config 'interface' 'loopback'
```

```
option 'ifname' 'lo'
option 'proto' 'static'
option 'ipaddr' '127.0.0.1'
option 'netmask' '255.0.0.0'
```

Berikut penjelasan detail konfigurasi *network* pada tabel 4.1. dan 4.2

1. Secara umum Konfigurasi *network* tersebut terdiri dari 2 bagian yaitu konfigurasi VLAN dan konfigurasi *interface*. Pada bagian konfigurasi VLAN dilakukan deklarasi VLAN serta pengaturan tentang jumlah vlan dan *port* yang digunakan oleh router. Sedangkan konfigurasi *interface*, mengatur tentang *virtual interface* yang aktif dan dapat digunakan oleh router.
2. *Config switch*: merupakan nama konfigurasi yang digunakan pada vlan untuk mendefinisikan *port* yang digunakan.
3. *Device*: merupakan perangkat *interface* sesungguhnya pada router yang akan digunakan. Pada Linksys digunakan "eth0" yang merupakan *interface* lan pada router dan pada TP-Link digunakan "rtl8366rb" yang merupakan type *interface* lan pada router
4. VLAN: merupakan nomor identitas/ID dari VLAN yang digunakan yang nantinya dipakai dalam penamaan *virtual interface*
5. *Port*: 0,1,2,3,4 merupakan *port* yang digunakan router pada VLAN sedangkan 5 atau 5t adalah nomor tag untuk mengakses eth0. Penomoran *port* pada konfigurasi, mempunyai urutan yang sama pada *port* yang ditampilkan router. Pada konfigurasi nomor *port* dimulai dari 0 sampai 4. Sedangkan pada tampilan *port*, nomor dimulai dari 1 sampai internet.
6. *Config interface*: merupakan konfigurasi *interface* untuk mendefinisikan *virtual interface* yang digunakan. Pada konfigurasi ini digunakan 4 buah VLAN dengan *port* yang berbeda beda yang terdiri atas eth0.0, eth0.1, eth0.2, eth0.3 pada Linksys dan eth0.1, eth0.2, eth0.3, eth0.4 pada TP-Link
7. *Ifname*: merupakan nama *virtual interface* pada router, dimana nama yang digunakan berdasarkan nama *interface* router dan nomor ID VLAN.
8. *Proto*: merupakan jenis pengalamatan pada *virtual interface* yang ada. Terdapat 2 jenis pengalamatan yaitu "static" dan "DHCP". Pengalamatan jenis static bersifat manual sedangkan DHCP bersifat otomatis. Pada jenis static perlu dilakukan penambahan "option ipaddr" dan "option netmask" untuk pengesetan alamat ip dan netmask

#### 4.2.2.3 Instalasi Dan Konfigurasi Openflow

Aplikasi Openflow memungkinkan router berfungsi sebagai sebuah SDN *switch* pada arsitektur jaringan SDN. Agar bisa bekerja sebagai SDN *switch*, Openflow pada router harus dikonfigurasi terlebih dahulu. Konfigurasi pada Openflow terdiri atas konfigurasi *port*, konfigurasi alamat ip *controller*, konfigurasi mode dan konfigurasi alamat ip. Berikut langkah-langkah instalasi dan konfigurasi Openflow

1. Salin source code Openflow pada github menggunakan perintah:

```
"git clone git://gitosis.stanford.edu/Openflow-Openwrt"
```

2. Masuk pada folder source yang telah di salin tadi, kemudian check versi dari Openflow sesuai versi router masing-masing. Untuk Linksys gunakan perintah: "git checkout -b Openflow-1.0/brcm origin/Openflow-1.0/brcm". Sedangkan untuk TP-Link gunakan perintah:

```
"git checkout -b Openflow-1.0/TP-Link origin/Openflow-1.0/TP-Link".
```

3. Masuk pada folder package, kemudian tambahkan extension Openflow dengan cara membuat symbolic link pada folder Openflow. Untuk linsys gunakan perintah:

```
"cd ~/ofwrt/backfire/packages/"
```

```
"ln -s ~/ofwrt/Openflow-Openwrt/Openflow-1.0/"
```

Sedangkan untuk TP-Link gunakan perintah:

```
"cd ~/Openwrt/trunk/package/"
```

```
"ln -s ~/Openwrt/Openflow-Openwrt/Openflow-1.0/"
```

4. Tambahkan konfigurasi dasar Openflow dengan membuat symbolic link pada folder diatas package yaitu folder "backfire" pada Linksys dan folder "trunk" pada TP-Link. Untuk linsys gunakan perintah:

```
"cd ~/ofwrt/backfire/"
```

```
"ln -s ~/ofwrt/Openflow-Openwrt/Openflow-1.0/files"
```

Sedangkan untuk TP-Link gunakan perintah:

```
"cd ~/Openwrt/trunk/"
```

```
"ln -s ~/Openwrt/Openflow-Openwrt/Openflow-1.0/files"
```

5. Masuk pada folder ofwrt kemudian build package tersebut dengan perintah: "make menuconfig".

6. Buka *file* konfigurasi Openflow dengan menggunakan perintah:

“vi /etc/config/Openflow” kemudian edit sesuai tabel 4.3

**Tabel 4.3 Konfigurasi Openflow**

```
config 'ofswitch'
  option 'dp' 'dp0'
  option 'ofports' 'eth0.0 eth0.1 eth0.2 eth0.3'
  option 'ofctl' 'tcp:192.168.2.10:6633'
  option 'mode' 'inband'
  option 'ipaddr' '192.168.2.1'
  option 'netmask' '255.255.255.0'
  option 'gateway' '192.168.2.1'
```

Berikut penjelasan dari konfigurasi Openflow pada tabel 3.3

1. *dp*: merupakan data path atau jalur yang digunakan router dalam mengirimkan paket
2. *ofports*: merupakan daftar Openflow *interface port* yang aktif dan dapat digunakan router. Nama *interface* eth0.0 eth0.1 eth0.2 eth0.3 mengacu pada konfigurasi *network* di router.
3. *ofctl*: merupakan definisi tentang Openflow kontroler yang berisikan konfigurasi alamat ip dan *port controller*. Alamat ip yang digunakan harus memiliki *network* yang berbeda dengan alamat bridge pada router. sedangkan *port* yang digunakan diatas merupakan *default port* dari cotroller Ryu
4. *mode*: mode Openflow yang digunakan ada 2 jenis, yaitu inband dan outband, Inband adalah pengesetan ip dan gateway jaringan dilakukan secara manual sedangkan outband adalah tanpa melakukan pengesetan ip dan gateway Openflow (*default*). Mode *inband* digunakan karena lebih mudah diimplementasikan dengan cara langsung mendeklarasikan alamat ip router. Sedangkan saat menggunakan *outband* terjadi kesulitan dalam mendeteksi *controller* yang ada.
5. *ipaddr*: merupakan alamat ip router sebagai SDN *switch*, penggunaan alamat ip harus pada *network* yang sama.
6. *netmask*: alamat *netmask* router sebagai Openflow oleh *switch*
7. *gateway*: alamat *gateway* router sebagai SDN *switch*

### 4.2.3 Application plane

Pada perancangan *application plane*, dilakukan modifikasi program untuk disesuaikan dengan proses pengujian. Proses penyesuaian program tersebut dilakukan pada saat pengujian waktu pembentukan *flow table* pada router. Modifikasi dilakukan dengan menambahkan keluaran pada *controller* berupa informasi waktu. Penambahan informasi waktu pada kode dilakukan menggunakan fungsi `time` pada python. Berikut potongan kode penambahan fungsi waktu pada aplikasi `simple_switch`.

**Tabel 4.4 Penambahan fungsi waktu pada `simple_switch`**

```
import time
def __init__(self, *args, **kwargs):
    super(SimpleSwitch, self).__init__(*args, **kwargs)
    self.mac_to_port = {}
    self.stt = time.time()

def _packet_in_handler(self, ev):
    ...
    if out_port != ofproto.OFPP_FLOOD:
        self.add_flow(datapath, msg.in_port, dst, actions)
        elapsed = time.time() - self.stt
        print elapsed
```

Pada tabel 5.1, awal waktu dideklarasikan dengan `stt` sedangkan waktu pembentukan *flow table* dideklarasikan dengan `elapsed`. `Stt` dideklarasikan pada fungsi `init` karena, pada fungsi tersebut proses pertama kali dimulai. Sedangkan `elapsed` ditaruh pada paket fungsi `packet in`, tepatnya pada kondisi pembentukan *flow table*. Nilai `elapsed` didapat dengan mengurangi `current time` dengan `stt`. Saat didapat nilai `elapsed`, nilai tersebut ditampilkan pada *controller* dengan satuan detik.

### 4.3 Pengujian

Pengujian dilakukan berdasarkan beberapa parameter uji yang telah ditentukan sebelumnya, yaitu sumberdaya komputasi dan jaringan. Kedua parameter tersebut dibagi menjadi beberapa bagian yaitu sebagai berikut.

1. Penggunaan *memory* dan CPU, mengetahui kemampuan router dalam menggunakan sumberdaya saat melakukan sebuah proses tertentu. Pengujian sumberdaya router tersebut diamati saat menggunakan Openflow dan tanpa menggunakan Openflow.
2. *Round trip time*, merupakan waktu yang dibutuhkan paket pada proses pengiriman untuk berpindah dari sumber ke tujuan dan kembali ke sumber lagi. Pengujian *round trip time* dilakukan dari *host* ke *host* lainnya saat menggunakan Openflow dan tanpa menggunakan Openflow.

3. *Throughput*, *Throughput* merupakan kecepatan pengiriman data pada waktu tertentu berdasarkan penggunaan bandwidth yang ada. Terdapat 2 jenis *throughput* yang diuji pada penelitian ini. Pengujian *throughput* pertama dilakukan pada pengiriman data antara *controller* dan router. Pengujian kedua dilakukan pada pengiriman *file* antar *host*
4. Waktu pembentukan *Flow table*, Pembentukan *flow table* pada router, digunakan untuk melihat respon time router terhadap instruksi yang diberikan *controller*.

#### 4.3.1 Skenario Pengujian dan Pengambilan data

##### 4.3.1.1 Penggunaan *Memory* dan CPU

Pengujian penggunaan *memory* dan CPU dilakukan dengan mengirimkan sebuah *file* dari *host* ke *host* lainnya menggunakan fungsi “scp”. Perintah “scp” atau *secure copy* merupakan perintah yang digunakan untuk mengirimkan *file* antar *host* pada suatu jaringan. Pengujian penggunaan *memory* dan CPU dilakukan pada Openflow-based router dan non-Openflow router. Dalam pengujian ini digunakan 2 macam variable yaitu: ukuran *file* dan jumlah *host* penerima. Ukuran *file* terdiri atas 500 MB, 1 GB dan 2 GB. Sedangkan jumlah *host* penerima terdiri atas 1 *host*, 2 *host*, 3 *host* dan 4 *host*. Pengujian berdasarkan ukuran *file* dilakukan dengan mengirimkan sebuah *file* dengan ukuran tertentu dari *host* 1 ke *host* 2/3/4. Apabila selesai dilakukan pengiriman *file* lagi dengan ukuran yang berbeda dari sebelumnya, dan begitu seterusnya. Pada pengujian berdasarkan variabel jumlah *host*, *file* tersebut dikirim dari *host* 1 ke 4 *host* lainnya menggunakan aplikasi Cluster ssh. Aplikasi Cluster ssh digunakan agar pengiriman *file* dapat dilakukan secara bersamaan. *File* dikirim dengan ukuran 100 MB dari *host* 1 ke *host* 2, *host* 1 ke *host* 2 dan 3, *host* 1 ke *host* 2,3 dan 4. Skema pengujian penggunaan *memory* dan CPU pada router dapat dilihat pada tabel 4.5 dan 4.6 berikut.

**Tabel 4.5 Pengujian penggunaan *memory* dan CPU pada router berdasarkan ukuran *file***

No	Ukuran <i>file</i>	Jumlah <i>host</i>
1	500 MB	1
2	1 GB	1
3	2 GB	1

**Tabel 4.6 Pengujian penggunaan *memory* dan CPU pada router berdasarkan jumlah *host***

No	Ukuran <i>file</i>	Jumlah <i>host</i>
1	100 MB	1
2	100 MB	2



3	100 MB	3
4	100 MB	4

Proses pengambilan data pada pengujian ini dilakukan menggunakan aplikasi Top yang ada pada router. Top merupakan aplikasi bawaan pada sistem operasi linux yang berfungsi menampilkan penggunaan *memory* dan CPU secara real time. Data penggunaan *memory* dan CPU saat proses pengiriman *file*, disimpan pada sebuah text *file* menggunakan aplikasi TOP. Penyimpanan data tersebut dilakukan dengan perintah “top -b >> ‘nama-file’ “. Data penggunaan *memory* dan CPU tersebut kemudian dihitung nilai rata-ratanya dari sampel data yang ada.

```
Mem: 23764K used, 5728K free, 0K shrd, 540K buff, 14356K cached
CPU:  35% usr  35% sys   0% nic   0% idle   0% io   0% irq  29% sirq
Load average: 0.83 1.56 1.01 2/35 1250
  PID  PPID  USER  STAT  VSZ %MEM %CPU COMMAND
  856    1 root    R   1280  4%  99% ofdatapath ptcp:6634 -i eth0.1 eth0.2
 1250  1159 root    R   1380  5%   0% top -b
 1159  1158 root    S   1392  5%   0% -ash
  309    1 root    S   1388  5%   0% syslogd -C16
    1    0 root    S   1384  5%   0% init
  295    1 root    S   1384  5%   0% init
 1155    1 root    S   1380  5%   0% /usr/sbin/ntpd -n -p 0.openwrt.pool.n
 1151    1 root    S   1380  5%   0% watchdog -t 5 /dev/watchdog
  297  294 root    S   1376  5%   0% logger -s -p 6 -t sysinit
  294    1 root    S   1376  5%   0% /bin/sh /etc/init.d/rcS S boot
  311    1 root    S   1368  5%   0% klogd
 1158  1111 root    S   1204  4%   0% /usr/sbin/dropbear -P /var/run/dropbe
 1111    1 root    S   1132  4%   0% /usr/sbin/dropbear -P /var/run/dropbe
  918    1 root    S   1092  4%   0% ofprotocol tcp:127.0.0.1:6634 tcp:192
 1141    1 nobody S    916  3%   0% /usr/sbin/dnsmasq -K -D -y -Z -b -E -
  325    1 root    S    792  3%   0% /sbin/hotplug2 --override --persisten
   92    2 root   SW     0  0%   0% [ar71xx-spi]
   42    2 root   SW     0  0%   0% [sync_supers]
  279    2 root   SWN    0  0%   0% [jffs2_gcd_mtd3]
    4    2 root   SW     0  0%   0% [events/0]
   72    2 root   SW     0  0%   0% [kswapd0]
   86    2 root   SW     0  0%   0% [mtdblockd]
    5    2 root   SW     0  0%   0% [khelper]
```

Gambar 4.6 Contoh data pengujian penggunaan *memory* dan CPU

#### 4.3.1.2 Round trip time

Proses pengujian *round trip time* menggunakan aplikasi ping yang dilakukan antar *host*. Dalam pengujian ini besarnya paket ping yang dikirimkan dijadikan variable penguji. Ukuran paket pada ping terdiri atas 64 bit, 512 bit, 1024 bit dan 2048 bit. Proses ping dilakukan dari *host* 1 ke *host* 2/3/4. Data pengujian *round trip time* diambil dari statistic hasil proses ping antar *host*. Data yang digunakan adalah *round trip time* min, max dan avg. Pengujian *round trip time* antar *host* dilakukan pada Openflow-based router dan non-Openflow router.



```

PING 192.168.1.32 (192.168.1.32) 2048(2076) bytes of data.
2056 bytes from 192.168.1.32: icmp_seq=1 ttl=64 time=2.99 ms
2056 bytes from 192.168.1.32: icmp_seq=2 ttl=64 time=1.83 ms
2056 bytes from 192.168.1.32: icmp_seq=3 ttl=64 time=1.79 ms
2056 bytes from 192.168.1.32: icmp_seq=4 ttl=64 time=1.75 ms
2056 bytes from 192.168.1.32: icmp_seq=5 ttl=64 time=1.54 ms
2056 bytes from 192.168.1.32: icmp_seq=6 ttl=64 time=1.68 ms
2056 bytes from 192.168.1.32: icmp_seq=7 ttl=64 time=1.40 ms
2056 bytes from 192.168.1.32: icmp_seq=8 ttl=64 time=1.42 ms
2056 bytes from 192.168.1.32: icmp_seq=9 ttl=64 time=1.42 ms
2056 bytes from 192.168.1.32: icmp_seq=10 ttl=64 time=1.78 ms
2056 bytes from 192.168.1.32: icmp_seq=11 ttl=64 time=1.93 ms
2056 bytes from 192.168.1.32: icmp_seq=12 ttl=64 time=1.77 ms
2056 bytes from 192.168.1.32: icmp_seq=13 ttl=64 time=1.43 ms
2056 bytes from 192.168.1.32: icmp_seq=14 ttl=64 time=1.67 ms
2056 bytes from 192.168.1.32: icmp_seq=15 ttl=64 time=1.75 ms
2056 bytes from 192.168.1.32: icmp_seq=16 ttl=64 time=1.41 ms
2056 bytes from 192.168.1.32: icmp_seq=17 ttl=64 time=1.71 ms
2056 bytes from 192.168.1.32: icmp_seq=18 ttl=64 time=1.68 ms
2056 bytes from 192.168.1.32: icmp_seq=19 ttl=64 time=1.86 ms
2056 bytes from 192.168.1.32: icmp_seq=20 ttl=64 time=1.90 ms

--- 192.168.1.32 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19035ms
rtt min/avg/max/mdev = 1.402/1.739/2.994/0.335 ms

```

Gambar 4.7 Contoh data pengujian *round trip time*

#### 4.3.1.3 Throughput

Pengujian *Throughput* antara *controller* ke router diuji menggunakan aplikasi *iftop*. *Iftop* merupakan aplikasi yang digunakan untuk mengamati bandwidth secara real time berdasarkan *host* dan *port*. Pengujian dilakukan saat aplikasi *simple switch* pertama kali dijalankan. Sehingga saat router pertama kali dihubungkan dengan *controller* dan aplikasi *simple switch* dijalankan, dilakukan pengukuran *throughput* dengan aplikasi *iftop*. Untuk mengetahui *throughput* antara *controller* ke router, *throughput* diidentifikasi berdasarkan alamat *host* dan *port* yang digunakan. Data *throughput* tersebut disimpan ke sebuah *file* dengan perintah "*iftop -tPn -i enp1s0 -l 2 >> 'nama-file'*". Kemudian data yang didapat, dihitung rata-ratanya dari keseluruhan data yang ada.

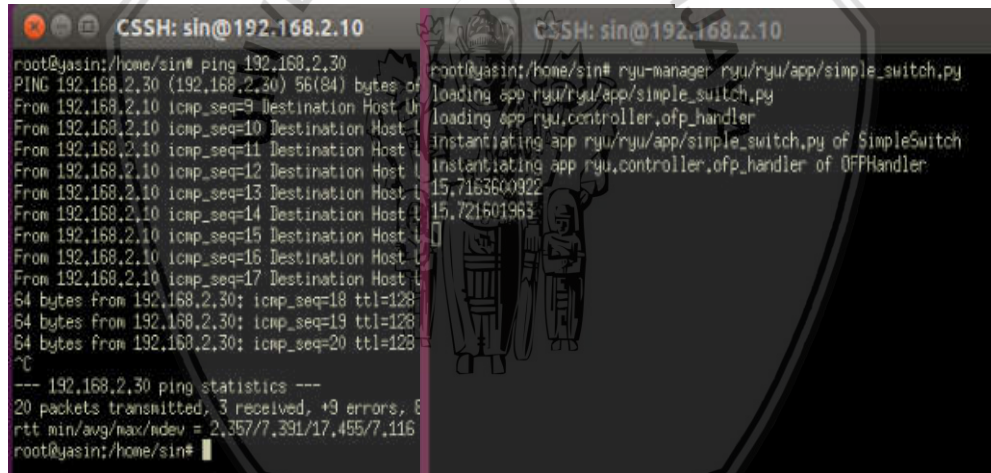
Listening on enp1s0					
#	Host name (port/service if enabled)		last 2s	last 10s	last 40s cumulative
-----					
Total send rate:			0b	0b	0b
Total receive rate:			0b	0b	0b
Total send and receive rate:			0b	0b	0b
-----					
Peak rate (sent/received/total):			0b	0b	0b
Cumulative (sent/received/total):			0B	0B	0B
=====					
#	Host name (port/service if enabled)		last 2s	last 10s	last 40s cumulative
-----					
1	192.168.2.10:6633	=>	4,33Kb	2,16Kb	2,16Kb
	192.168.2.1:59857	<=	5,45Kb	2,72Kb	2,72Kb
2	192.168.2.10	=>	0,98Kb	504b	504b
	192.168.2.30	<=	0,98Kb	504b	504b
-----					
Total send rate:			5,31Kb	2,66Kb	2,66Kb
Total receive rate:			6,43Kb	3,21Kb	3,21Kb
Total send and receive rate:			11,7Kb	5,87Kb	5,87Kb
-----					
Peak rate (sent/received/total):			5,31Kb	6,43Kb	11,7Kb
Cumulative (sent/received/total):			1,33KB	1,61KB	2,94KB
=====					

Gambar 4.8 Contoh data pengujian *throughput*

Pengujian *throughput* antar *host* dilakukan dengan mengirimkan sebuah *file* dari satu *host* ke *host* lain menggunakan fungsi “*scp*”. *File* yang dikirimkan berukuran 100 MB. Proses pengambilan data dilakukan dengan mengamati proses pengiriman *file* pada terminal, terkait *throughput* dan lamanya waktu pengiriman. Pengujian *throughput* antar *host* dilakukan pada Openflow-based router dan non-Openflow router.

#### 4.3.1.4 Waktu Pembentukan *Flow table*

Proses pengujian dilakukan dengan mengaktifkan aplikasi *simple switch* bersamaan dengan dilakukannya proses ping ke *host* lain. Hal tersebut dilakukan disaat *host* sesaat setelah *host* terkoneksi dengan switch. Pada saat proses ping mendapat reply dari *host* yang dituju, hal tersebut menandakan bahwa *flow table* telah terbentuk. Aplikasi *simple switch* akan memberikan keluaran berupa angka. Pada saat *flow table* terbentuk. Angka tersebut merupakan durasi terbentuknya *flow table* yang di tuliskan dengan satuan detik. Untuk mendapatkan hasil yang lebih akurat, pengujian tersebut dilakukan sebanyak 3 kali. Aplikasi *simple switch* dan ping dijalankan bersamaan menggunakan aplikasi CSSH. Pada pengujian ini, data yang diambil berasal dari nilai rata-rata pada keluaran aplikasi *simple switch*.



```

CSSH: sin@192.168.2.10
root@yasini:/home/sin# ping 192.168.2.30
PING 192.168.2.30 (192.168.2.30) 56(84) bytes of data:
From 192.168.2.10 icmp_seq=9 Destination Host Unreachable
From 192.168.2.10 icmp_seq=10 Destination Host Unreachable
From 192.168.2.10 icmp_seq=11 Destination Host Unreachable
From 192.168.2.10 icmp_seq=12 Destination Host Unreachable
From 192.168.2.10 icmp_seq=13 Destination Host Unreachable
From 192.168.2.10 icmp_seq=14 Destination Host Unreachable
From 192.168.2.10 icmp_seq=15 Destination Host Unreachable
From 192.168.2.10 icmp_seq=16 Destination Host Unreachable
From 192.168.2.10 icmp_seq=17 Destination Host Unreachable
64 bytes from 192.168.2.30: icmp_seq=18 ttl=128: time=0.000ms
64 bytes from 192.168.2.30: icmp_seq=19 ttl=128: time=0.000ms
64 bytes from 192.168.2.30: icmp_seq=20 ttl=128: time=0.000ms
^C
--- 192.168.2.30 ping statistics ---
20 packets transmitted, 3 received, +9 errors, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 2.357/7.391/17.455/7.116
root@yasini:/home/sin#

CSSH: sin@192.168.2.10
root@yasini:/home/sin# ryu-manager ryu/ryu/app/simple_switch.py
loading app ryu/ryu/app/simple_switch.py
loading app ryu.controller.ofp_handler
Instantiating app ryu/ryu/app/simple_switch.py of SimpleSwitch
Instantiating app ryu.controller.ofp_handler of OFPHandler
15.7163600922
15.721501963

```

Gambar 4.9 Pengujian waktu pembentukan *flow table* pada *controller*

## BAB 5 HASIL DAN PEMBAHASAN

### 5.1 Penggunaan *memory* dan CPU

CPU dan *Memory* merupakan komponen penting dalam sebuah router dan dapat dijadikan sebagai acuan dalam menentukan kinerja router tersebut. Pada penelitian ini dilakukan pengujian mengenai besarnya konsumsi sumberdaya CPU dan *memory* router saat menggunakan Openflow, serta untuk mengetahui pengaruh dari penggunaan Openflow terhadap sumberdaya tersebut. Berikut hasil pengujian penggunaan *memory* dan CPU pada router Linksys WRT54GL dan TP-Link TL-WR1043ND pada arsitektur SDN dan tradisional.

**Tabel 5.1 Hasil pengujian penggunaan CPU dan *memory* saat router kondisi idle**

Router	Enable Openflow	Memory	CPU
TP-Link WR1403ND	Ya	16508 KB	0%
	Tidak	16504 KB	0%
Linksys WRT54GL	Ya	11980 KB	1%
	Tidak	11292 KB	1%

Berdasarkan tabel diatas, penerapan Openflow pada saat kondisi router idle tidak memberi pengaruh yang signifikan terhadap penggunaan CPU dan *memory* router. Pada router TP-Link ataupun Linksys, penggunaan Openflow tidak terlihat menambah konsumsi sumberdaya CPU. Penggunaan CPU pada router TP-Link mempunyai nilai kurang dari 1% dan pada router Linksys bernilai 1%. Sedangkan pada penggunaan *memory*, penggunaan Openflow dapat meningkatkan konsumsi sumberdaya *memory* pada router. Penggunaan sumberdaya *memory* pada router TP-Link meningkatkan konsumsi *memory* sebesar 4 KB dan router Linksys sebesar 688 KB.

**Tabel 5.2 Hasil pengujian penggunaan CPU dan *memory* pada Linksys WRT54GL berdasarkan ukuran *file***

Ukuran <i>file</i>	Jumlah <i>host</i>	Alamat ip <i>host</i>	Penggunaan rata-rata <i>memory</i>		Penggunaan rata-rata CPU	
			Non Openflow	<i>Enable</i> Openflow	Non Openflow	<i>Enable</i> Openflow
500 MB	1	192.168.1.32	11816 KB	12082 KB	75%	96%
1 GB	1	192.168.1.32	11821 KB	12714 KB	81%	96%
2 GB	1	192.168.1.32	11829 KB	12745 KB	82%	97%

Pada tabel 5.2 diatas, terlihat nilai penggunaan sumberdaya CPU dan memory terus bertambah apabila ukuran *file* yang dikirim semakin besar, baik pada saat penggunaan Openflow ataupun tidak. Hal tersebut menunjukkan bahwa semakin besar *file* yang dikirim semakin besar pula sumberdaya CPU dan memory yang digunakan.

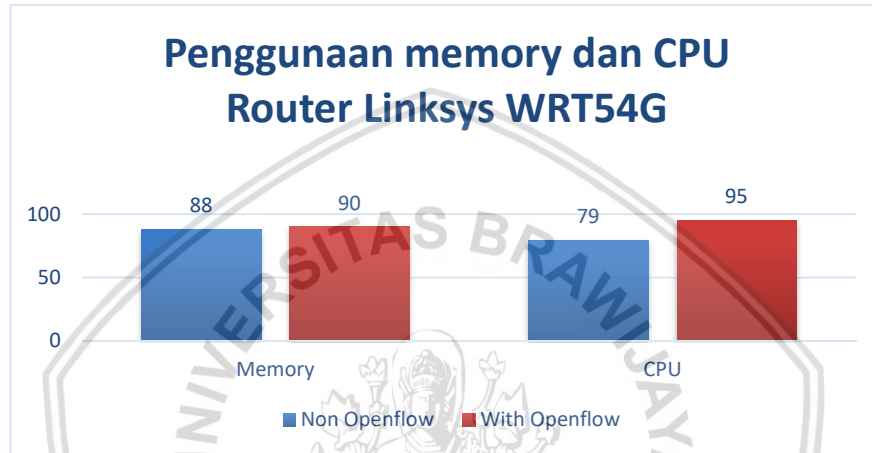
**Tabel 5.3 Hasil pengujian penggunaan CPU dan *memory* pada Linksys WRT54GL berdasarkan jumlah *host***

Ukuran <i>file</i>	Jumlah <i>host</i>	Alamat ip <i>host</i>	Penggunaan rata-rata <i>memory</i>		Penggunaan rata-rata CPU	
			Non Openflow	<i>Enable</i> Openflow	Non Openflow	<i>Enable</i> Openflow
100 MB	1	192.168.1.32	11798 KB	11854 KB	66%	92%
100 MB	2	192.168.1.32	11817 KB	11859 KB	71%	93%
		192.168.1.33				
100 MB	3	192.168.1.32	11871 KB	11930KB	82%	95%
		192.168.1.33				
		192.168.1.33				
100 MB	4	192.168.1.32	11892 KB	12088 KB	82%	95%
		192.168.1.33				
		192.168.1.34				
		192.168.1.35				

Pada tabel 5.3 diatas, terlihat nilai penggunaan sumberdaya CPU dan memory terus bertambah apabila jumlah *host* yang dituju semakin banyak, baik pada saat penggunaan Openflow ataupun tidak. Hal tersebut menunjukkan bahwa semakin banyak jumlah *host* yang dituju semakin besar pula sumberdaya CPU dan memory yang digunakan.

Tabel 5.4 Rata-rata penggunaan *memory* dan CPU router Linksys WRT54G

<i>Enable Openflow</i>	<i>Memory</i>	Presentase <i>Memory</i>	Presentase CPU
Tidak	11835 KB	88%	79%
Ya	12180 KB	90%	95%

Gambar 5.1 Rata-rata penggunaan *memory* dan CPU router Linksys

Berdasarkan gambar 5.1 menunjukkan bahwa, penggunaan Openflow dapat meningkatkan penggunaan sumberdaya CPU dan memory pada router Linksys WRT54GL. Rata-rata peningkatan penggunaan sumberdaya CPU sebesar 16% dan memory sebesar 345 KB atau sekitar 2% dari total memory. Penggunaan sumberdaya CPU dan memory pada router saat menggunakan Openflow hampir mendekati 100% pada setiap pengiriman *file*. Berdasarkan hal tersebut, dapat diambil kesimpulan bahwa penggunaan Openflow pada router Linksys WRT54GL membutuhkan sumberdaya yang banyak.

**Tabel 5.5 Hasil pengujian penggunaan CPU dan *memory* pada TP-Link WR1043ND Openflow berdasarkan ukuran *file***

Ukuran <i>file</i>	Jumlah <i>host</i>	Alamat ip <i>host</i>	Penggunaan rata-rata <i>memory</i>		Penggunaan rata-rata CPU	
			Non Openflow	Enable Openflow	Non Openflow	Enable Openflow
500 MB	1	192.168.1.32	16522 KB	23776 KB	18%	89%
1 GB MB	1	192.168.1.32	16552 KB	23896 KB	25%	91%
2 GB MB	1	192.168.1.32	16671 KB	24136 KB	26%	92%

Seperti halnya pada router Linksys, pada tabel 5.5 diatas, menunjukkan bahwa semakin besar *file* yang dikirim semakin besar pula sumberdaya CPU dan memory yang digunakan. Pada router TP-Link, penggunaan sumberdaya CPU saat tidak menggunakan Openflow relative kecil. Penggunaan sumberdaya CPU yang kecil tersebut terlihat meningkat drastis ketika router menggunakan Openflow.

**Tabel 5.6 Hasil pengujian penggunaan CPU dan *memory* TP-Link WR1043ND berdasarkan jumlah *host***

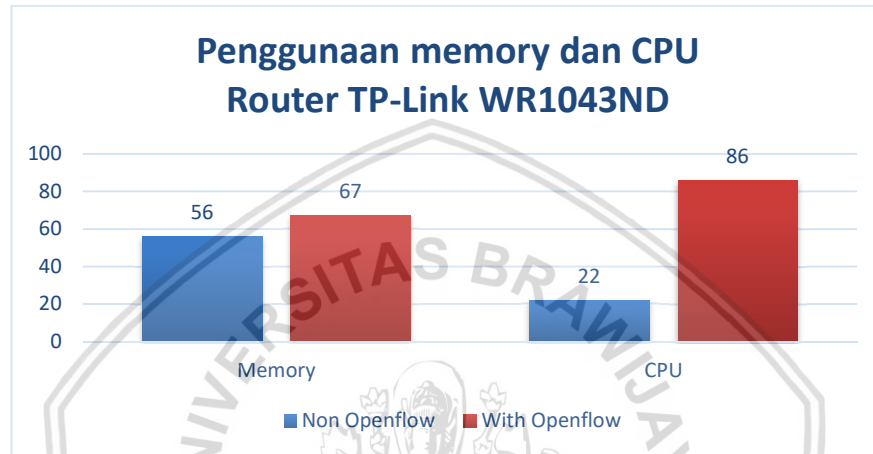
Ukuran <i>file</i>	Jumlah <i>host</i>	Alamat ip <i>host</i>	Penggunaan rata-rata <i>memory</i>		Penggunaan rata-rata CPU	
			Non Openflow	Enable Openflow	Non Openflow	Enable Openflow
100 MB	1	192.168.1.32	16511 KB	16605 KB	15%	73%
100 MB	2	192.168.1.32	16516 KB	16638 KB	27%	83%
		192.168.1.33				
100 MB	3	192.168.1.32	16539 KB	16655 KB	23%	89%
		192.168.1.33				
		192.168.1.33				
100 MB	4	192.168.1.32	16570 KB	16702 KB	24%	91%
		192.168.1.33				
		192.168.1.34				
		192.168.1.35				

Seperti halnya pada router Linksys, pada tabel 5.6 diatas, menunjukkan bahwa semakin banyak *host* yang dituju semakin besar pula sumberdaya CPU dan memory yang digunakan.



Tabel 5.7 Rata-rata penggunaan *memory* dan CPU router TP-Link TL-WR1043ND

<i>Enable Openflow</i>	<i>Memory</i>	Presentase <i>Memory</i>	Presentase CPU
Tidak	16554 KB	56%	22%
Ya	19773 KB	67%	86%

Gambar 5.2 Rata-rata penggunaan *memory* dan CPU router Linksys

Berdasarkan gambar 5.2 menunjukkan bahwa, penggunaan Openflow dapat meningkatkan penggunaan sumberdaya CPU dan memory pada router TP-Link TL-WR1043ND. Rata-rata peningkatan penggunaan sumberdaya CPU sebesar 64% dan memory sebesar 3219 KB atau sekitar 11% dari total memory. Penggunaan sumberdaya CPU dan memory pada router saat menggunakan Openflow hampir mendekati 100% pada setiap pengiriman *file*. Berdasarkan hal tersebut, dapat diambil kesimpulan bahwa penggunaan Openflow pada router TP-Link TL-WR1043ND juga membutuhkan sumberdaya yang banyak.

Saat menjalankan fungsi Openflow, router masih menjalankan fungsi sebagai *legacy/regular switch* sehingga penggunaan CPU dan *memory* menjadi lebih besar. Pada arsitektur SDN terdapat proses tambahan yang dijalankan router agar dapat berfungsi sebagai SDN *switch*. Router menjalankan proses ofprotocol dan ofdatapath saat berperan sebagai SDN *switch*. Ofprotocol dan Ofdatapath merupakan bagian dari aplikasi Openflow. Ofprotocol berfungsi sebagai sarana komunikasi antara router dan *controller*. Sedangkan ofdatapath berfungsi sebagai penghubung antar *host*.

## 5.2 Round trip time

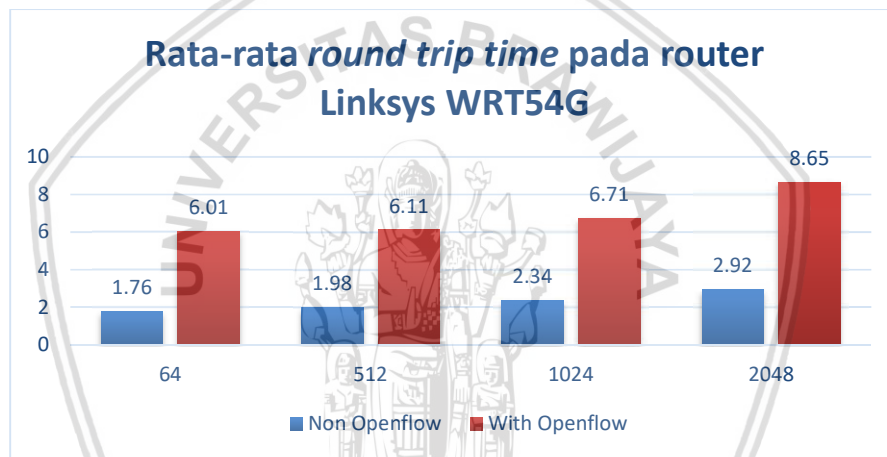
Untuk melihat kualitas koneksi jaringan pada router dan *host*, bisa dilakukan dengan mengamati nilai *round trip time* dari sumber ke tujuan. Secara umum apabila nilai *round trip time* nya tinggi, berarti kondisi koneksi jaringan tersebut kurang bagus dan juga sebaliknya. Dalam penelitian ini, dilakukan pengujian pada



2 router dengan arsitektur jaringan tradisional dan SDN. Berikut hasil pengujian *round trip time* router Linksys WRT54GL dan TP-Link TL-WR1043ND pada arsitektur SDN dan tradisional.

**Tabel 5.8 Hasil pengujian *round trip time* Linksys WRT54GL**

Besar paket	<i>round trip time</i> minimal		<i>round trip time</i> maksimal		<i>round trip time</i> rata-rata	
	Non Openflow	Enable Openflow	Non Openflow	Enable Openflow	Non Openflow	Enable Openflow
64	1,31 ms	5,85 ms	3,81 ms	6,14 ms	1,76 ms	6,01 ms
512	1,59 ms	2,77 ms	2,42 ms	8,09 ms	1,98 ms	6,11 ms
1024	2,17 ms	3,26 ms	2,59 ms	7,29 ms	2,34 ms	6,71 ms
2048	2,41 ms	5,84 ms	3,60 ms	9,16 ms	2,92 ms	8,65 ms

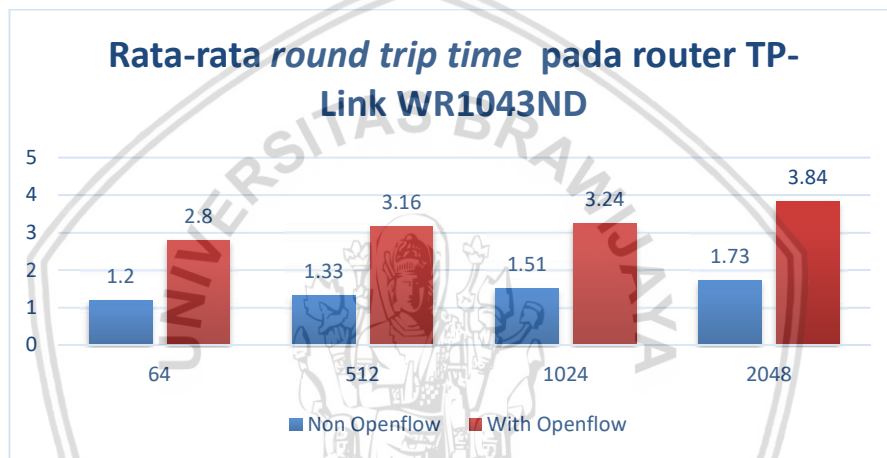


**Gambar 5.3 Rata-rata *round trip time* pada router Linksys WRT54G**

Pada gambar 5.2 diatas, didapat nilai *round trip time* rata-rata router Linksys WRT54GL sebesar 2,02 ms pada arsitektur tradisional dan arsitektur SDN sebesar 6,87 ms. Berdasarkan hal tersebut menunjukkan bahwa, penggunaan Openflow dapat meningkatkan nilai *round trip time* router Linksys WRT54GL. Rata-rata peningkatan nilai *round trip time* sebesar 4,61 ms.

**Tabel 5.9 Hasil pengujian *round trip time* TP-Link WR1043ND**

Besar paket	<i>round trip time</i> minimal		ROUND TRIP TIME maksimal		ROUND TRIP TIME rata-rata	
	Non Openflow	Enable Openflow	Non Openflow	Enable Openflow	Non Openflow	Enable Openflow
64	0,77 ms	1,54 ms	1,59 ms	3,24 ms	1,20 ms	2,84 ms
512	1,06 ms	2,94 ms	1,59 ms	3,32 ms	1,33 ms	3,16 ms
1024	1,14 ms	2,11 ms	1,91 ms	3,48 ms	1,51 ms	3,24 ms
2048	1,40 ms	2,35 ms	2,99 ms	5,91 ms	1,73 ms	3,84 ms



**Gambar 5.4 Rata-rata *round trip time* pada router TP-Link WR103ND**

Router TP-Link memiliki nilai *round trip time* rata-rata sebesar 1,4 ms pada arsitektur tradisional dan 3,27 pada arsitektur SDN. Berdasarkan hal tersebut menunjukkan bahwa, penggunaan Openflow dapat meningkatkan nilai *round trip time* router TP-Link WR103ND. Rata-rata peningkatan nilai *round trip time* sebesar 1,56 ms.

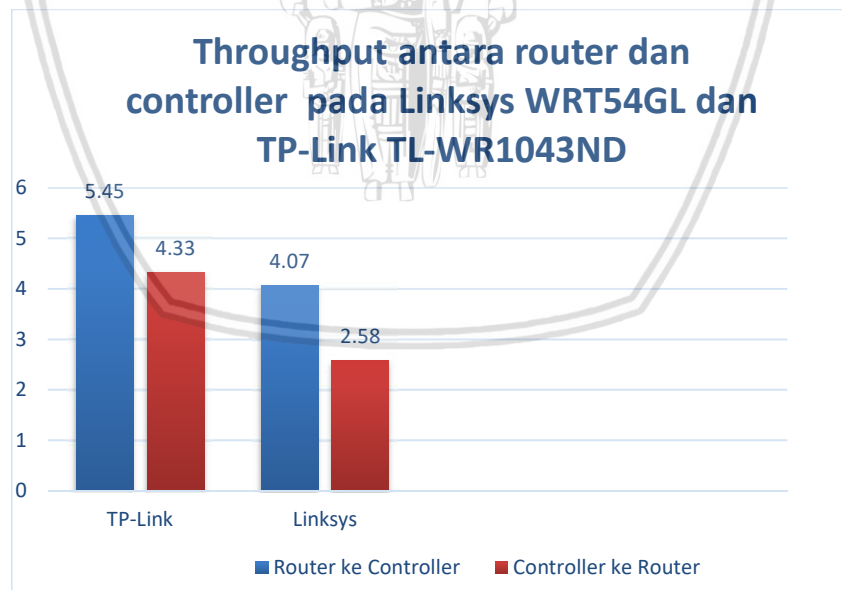
Penggunaan router sebagai SDN *switch* dengan menambahkan fungsi Openflow dikenal dengan istilah *hybrid Openflow switch*. Pada *hybrid Openflow switch*, fungsi *legacy/regular switch* tetap dijalankan saat fungsi Openflow aktif. Saat *switch* mengirim sebuah data, terdapat mekanisme untuk menentukan bagaimana paket tersebut akan diproses. Apakah paket diproses menggunakan Openflow atau diproses menggunakan *legacy/tradisional*. Penambahan Openflow pada router berarti menambah proses yang dijalankan oleh router, sehingga router membutuhkan waktu yang lebih banyak untuk menjalankan proses tambahan tersebut. Oleh karena, itu nilai *round trip time* pada router meningkatkan.

### 5.3 Throughput

*Throughput* merupakan parameter yang umum dijadikan acuan dalam menentukan kualitas koneksi jaringan. Semakin tinggi nilai *throughput* semakin bagus juga kualitas jaringannya. *Throughput* antara router dan *controller* dapat dijadikan acuan dalam melihat kinerja Openflow dalam menghubungkan antara router dan *controller*. Sedangkan *throughput* antar *host* dijadikan acuan terkait seberapa cepat router dalam mengkomunikasikan antar *host* pada suatu jaringan. Berikut hasil pengujian *throughput* antara router dan *controller* pada saat pembentukan *flow table*.

**Tabel 5.10 Hasil pengujian *throughput* antara router dan *controller* pada Linksys WRT54GL dan TP-Link TL-WR1043ND**

Router	<i>Throughput</i>	
	Router ke <i>controller</i>	<i>Controller</i> ke router
Linksys WRT54GL	4,07 Kb/s	2,58 Kb/s
TP-Link TL-WR1043ND	5,45 Kb/s	4,33 Kb/s



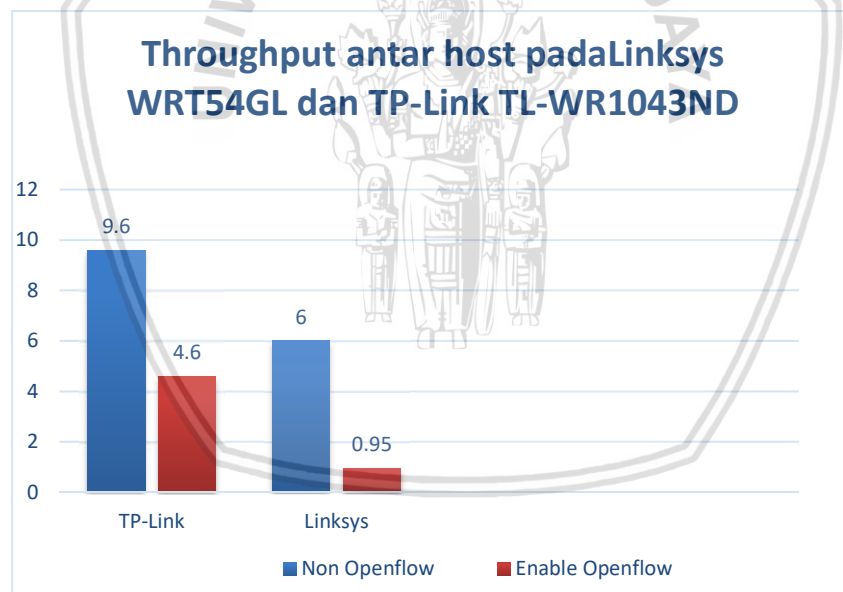
**Gambar 5.5 Grafik *throughput* antara router dan *controller* pada Linksys WRT54GL dan TP-Link TL-WR1043ND**

Pada SDN *switch*, *controller* mengirimkan data ke router hanya saat router tidak mempunyai informasi paket pada *flow* entries atau paket mengharuskan

diproses di kontroller. Apabila router telah memiliki informasi tersebut, router akan langsung mengeksekusikan tindakan terhadap paket yang ada, berdasarkan informasi *flow* entries yang didapat dari *controller* sebelumnya. Data yang dikirimkan *controller* ke router berupa header field dari sebuah Data. Data header field ini kemudian, dicocokkan pada entri yang ada untuk menentukan aksi terhadap suatu paket. Kecepatan *throughput* yang kecil tersebut disebabkan karena header field yang diforward ke *controller* memiliki ukuran yang kecil juga.

**Tabel 5.11 Hasil pengujian *throughput* antar *host* pada Linksys WRT54GL dan TP-Link TL-WR1043ND**

Router	<i>Throughput</i>	
	<i>Enable Openflow</i>	Non Openflow
Linksys WRT54GL	0,95 MB/s	6 MB/s
TP-Link TL-WR1043ND	4,6 MB/s	9,6 MB/s



**Gambar 5.6 Grafik *throughput* antar *host* pada Linksys WRT54GL dan TP-Link TL-WR1043ND**

Berdasarkan gambar 5.6 diatas, menunjukkan bahwa penggunaan Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND dapat menurunkan *throughput*. Penggunaan Openflow pada router TP-Link menurunkan *throughput* sebesar 5 MB/s atau sekitar 52% dibanding saat tidak menggunakan Openflow. Sedangkan pada router Linksys, *throughput* turun sebesar 5,05 MB/s atau sekitar 84% dibanding saat tidak menggunakan Openflow. Saat ditambahkan Openflow,

router mengalami penambahan waktu pemrosesan forwarding paket/data akibat mekanisme *hybrid* Openflow *switch* serta tingginya konsumsi CPU dan *memory* untuk proses Openflow menyebabkan turunnya kemampuan *switching* router. Hal tersebut mengakibatkan nilai throughput menurun.

#### 5.4 Waktu pembentukan *flow table*

Pada arsitektur SDN, *switch* baru dapat berfungsi sesuai yang diprogramkan apabila *switch* telah mengolah instruksi yang di terima dari *controller*. Instruksi tersebut adalah membentuk sebuah *flow table* agar *host* yang ada dapat terhubung satu sama lain dan router dapat memforward paket berdasarkan *flow* tabel tersebut. Berikut hasil pengujian waktu pembentukan *flow table* pada Linksys WRT54GL dan TP-Link TL-WR1043ND pada arsitektur SDN.

**Tabel 5.12 Hasil pengujian waktu pembentukan *flow table* router Linksys WRT54GL**

Jenis router	Pengujian ke -	Waktu
Linksys WRT54GL	1	15,7
	2	15,6
	3	10,5
	4	16,7
	5	21,7
	6	14,6
	7	15,7
	8	15,6
	9	14,6
	10	16,7
Rata-rata		15,74

**Tabel 5.13 Hasil pengujian waktu pembentukan *flow table* router TP-Link TL-WR1043ND**

Jenis router	Pengujian ke -	Waktu
TP-Link TL-WR1043ND	1	14,6
	2	15,7
	3	15,7
	4	10,5
	5	16,7
	6	7,6
	7	14,6
	8	15,6
	9	14,6
	10	15,7
Rata-rata		14.13

Berdasarkan tabel 5.12 dan tabel 5.1, didapatkan nilai rata-rata pembentukan *flow table* pada router Linksys WRT54GL sebesar 15,74 detik dan router Tp-Link TL-WR1043ND sebesar 14,13 detik. Hasil waktu yang didapat pada masing-masing router menunjukkan seberapa cepat router sebagai SDN *switch* memproses perintah dari *controller*. Hasil pembentukan *flow table* dipengaruhi oleh kecepatan throughput pada masing-masing perangkat. Dalam hal ini, throughput yang terkait adalah throughput antara router dan *controller*, bukan throughput antar host. Hal tersebut dapat diketahui dengan melihat hasil waktu pembentukan *flow table* pada masing-masing perangkat yang mempunyai selisih tidak jauh. Dengan tingginya *throughput* antara SDN *switch* dan *controller*, membuat proses pembentukan *flow table* berjalan lebih cepat

## BAB 6 KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Berdasarkan pengujian dan analisis kinerja Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND yang telah dilakukan, maka diperoleh beberapa kesimpulan sebagai berikut.

1. Dalam merancang lingkungan pengujian kinerja Openflow-based switch pada router Linksys WRT54GL dan TP-Link TL-WR1043ND pada arsitektur SDN terdapat 4 tahapan yang harus dilakukan. Tahapan perancangan tersebut meliputi, topologi jaringan, *control plane*, *data plane*, dan *application plane*. Perancangan topologi merupakan tahapan pertama yang harus dilakukan, dalam tahapan ini perangkat yang ada disusun menggunakan topologi linier yang dianggap paling efektif untuk pengujian. Berikutnya tahapan perancangan *control plane* atau aplikasi Ryu. Perancangan pada *control plane* meliputi instalasi dan konfigurasi aplikasi Ryu. Tahapan selanjutnya adalah perancangan *data plane* atau router. Tahapan perancangan pada *data plane* meliputi instalasi *firmware* Openwrt, instalasi Openflow dan konfigurasi router. Tahapan terakhir adalah perancangan *application plane*, dalam hal ini digunakan program *simple switch* yang diambil dari *library* Ryu. Pada perancangan *application plane*, dilakukan modifikasi program untuk disesuaikan dengan proses pengujian.
2. Pada pengujian penggunaan sumberdaya CPU dan *memory* menggunakan Openflow, router Linksys WRT54GL menggunakan sumberdaya rata-rata CPU sebesar 90% dan *memory* sebesar 12180 KB atau sekitar 95% dari jumlah total *memory*. Kemudian pada pengujian *round trip time*, didapat rata-rata nilai *round trip time* sebesar 6.87 ms. Pada pengujian *throughput* antara router dan *controller* didapat nilai 4,07 KB/s pada saat pengiriman data dari router ke *controller* dan 2.58 KB/s pada saat pengiriman data dari *controller* ke router. Sedangkan untuk *throughput* antar *host* didapat nilai 0.95 MB/s. Selanjutnya pada pengujian waktu pembentukan *flow table* didapat hasil sebesar 16 detik. Pada pengujian penggunaan sumberdaya CPU dan *memory* menggunakan Openflow, router TP-Link TL-WR1043ND menggunakan sumberdaya rata-rata CPU sebesar 86% dan *memory* sebesar 19773 KB atau sekitar 67% dari jumlah total *memory*. Kemudian pada pengujian *round trip time*, didapat rata-rata nilai *round trip time* sebesar 3.27 ms. Pada pengujian *throughput* antara router dan *controller* didapat nilai 5,45 KB/s pada saat pengiriman data dari router ke *controller* dan 4,33 KB/s pada saat pengiriman data dari *controller* ke router. Sedangkan untuk *throughput* antar *host* didapat nilai 4,6 MB/s. Selanjutnya pada pengujian waktu pembentukan *flow table* didapat hasil sebesar 15.3 detik.



3. Penambahan Openflow pada router Linksys WRT54GL dan TP-Link TL-WR1043ND yang diinstal Openwrt dapat meningkatkan penggunaan sumberdaya CPU dan *memory* serta dapat menurunkan kinerja jaringan seperti *round trip time* dan *throughput*. Penurunan kinerja router tersebut disebabkan penambahan proses Openflow pada router yang mengakibatkan tingginya konsumsi memory dan CPU sehingga kemampuan switching menurun. Selain itu mekanisme dari *hybrid* Openflow switch dapat menyebabkan penambahan waktu pemrosesan saat proses *forwarding* paket/data.

## 6.2 Saran

Terdapat beberapa saran untuk pengembangan penelitian ini kedepannya, berdasarkan hasil penelitian yang telah dilakukan, sebagai berikut

- 1 Perlu dilakukan perancangan sendiri algoritma program pada *controller* untuk mendapatkan hasil sesuai tujuan yang diinginkan
- 2 Perlu dilakukan metode pengujian yang berbeda untuk melihat penggunaan sumberdaya router yang lebih detail
- 3 Mengganti topologi yang digunakan ke topologi yang lebih kompleks untuk mendapatkan hasil yang lebih baik. Semakin kompleks topologi yang digunakan semakin banyak hasil yang didapatkan

## DAFTAR PUSTAKA

- Abdillah, N. (2016). ANALISIS PERFORMA ARSITEKTUR SOFTWARE DEFINED NETWORK DENGAN OPENFLOW PADA MIKROTIK RB750. *Naskah Publikasi*, 8.
- Aris Cahyadi Risdianto, M. A. (2014, september 10). *Kontroler opendaylight*. Diambil ulang september 10, 2016
- Aris Cahyadi Risdianto, M. A. (2014, september 10). *Pengantar SDN*. Diambil ulang september 10, 2016
- Bakhshi, T. (2017). State of the Art and Recent Research Advances in Software Defined Networking. *Wireless Communications and Mobile Computing*.
- bps.go.id. (2016). *INDEKS PEMBANGUNAN TEKNOLOGI INFORMASI DAN KOMUNIKASI*. Diambil ulang 11 23, 2017, from [https://bps.go.id/website/brs\\_ind/brsInd-20161215131927.pdf](https://bps.go.id/website/brs_ind/brsInd-20161215131927.pdf)
- Budiyanto, a. (2012). *Pengantar Cloud Computing* (1 ed.). jakarta: CloudIndonesia.ORG.
- Detik.com. (2013). *Router murah*. Diambil ulang 12 16, 2017, from <http://forum.detik.com/pusat-belanja-router-usb-wifi-tplink-termurah-t626404.html>
- Doyle, L. (2013). *Does software-defined networking require special SDN switches*. Diambil ulang 12 29, 2017, from <http://searchsdn.techtarget.com/answer/Does-software-defined-networking-require-special-SDN-switches>
- Goransson, C. B. (2014). *Software Defined Networks: A Comprehensive Approach*. USA: Google.
- Hidayat, M. H. (2017). *IMPLEMENTASI DAN ANALISIS KINERJA ARSITEKTUR SOFTWARE DEFINED NETWORK BERBASIS OPENDAYLIGHT CONTROLLER*. YOGYAKARTA: UNIVERSITAS GADJAH MADA.
- Idris Z. Bholebawa, R. K. (2016). Performance Analysis of Proposed Network. *International Journal of Computer Science and Information Security (IJCSIS)*, 11.
- inet.detik.com. (2012). *inet.detik.com*. Diambil ulang 12 16, 2017, from <https://inet.detik.com/business/d-2099068/kompleksitas-data-center-bikin-repot>

- Kanika. (2016). *SDN Tutorials*. Diambil ulang 12 28, 2017, from <http://sdntutorials.com/difference-between-control-plane-and-data-plane/>
- Mardiansyah, A. (2014). *Apa itu network engineer*. Diambil ulang september 9, 2016, from <http://achmad.glcnetworks.com/2014/06/30/apa-itu-network-engineer-program-studi-apa-yang-dapat-menjadikan-network-engineer/>
- Mitchell, B. (2017). *What Is a Router for Computer Networks*. Diambil ulang 11 23, 2017, from <https://www.lifewire.com/how-routers-work-816456>
- mulyana, E. (2014). *Pengantar SDN*. Diambil ulang from eueung.gitbooks.io: [https://eueung.gitbooks.io/buku-komunitas-sdn-rg/content/pengantar\\_sdn/README.html](https://eueung.gitbooks.io/buku-komunitas-sdn-rg/content/pengantar_sdn/README.html)
- Open Fondation, N. (2017). *Software-Defined Networking (SDN) Definition*. Diambil ulang November 23, 2017, from <https://www.opennetworking.org/sdn-definition/>
- Open Networking Foundation. (2016). *Openflow*. Diambil ulang from Open Networking Foundation: <https://www.opennetworking.org/sdn-resources/Openflow>
- Open Networking, F. (2011). *Learn more Openflow*. Diambil ulang 11 23, 2017, from <http://archive.Openflow.org/wp/learnmore/>
- Openwrt. (2013). *Openwrt*. Diambil ulang 12 29, 2017, from [Openwrt.org](http://Openwrt.org)
- Prahadi, Y. Y. (2015). *swa.co.id*. Diambil ulang 12 15, 2017, from <https://swa.co.id/swa/headline/pwc-perkembangan-cloud-computing-di-indonesia-lamban>
- Rikie Kartadie, T. S. (2015). *UJI PERFORMA SOFTWARE-BASED OPENFLOW. Uji Performa Software Based Openflow Switch*, 13.
- SDXcentral. (2014). *What is Ryu controller*. Diambil ulang September 2, 2017, from <https://www.sdxcentral.com:https://www.sdxcentral.com/sdn/definitions/sdn-controllers/open-source-sdn-controllers/what-is-Ryu-controller/>
- Teachtarget. (2015). *Facebook, Google use SDN to boost data center connectivity*. Diambil ulang 12 15, 2017, from <http://searchsdn.techtarget.com/tip/Facebook-Google-use-SDN-to-boost-data-center-connectivity>